

**SISTEM PENGKLASIFIKASI PENENTUAN JURUSAN KULIAH  
UNTUK PELAJAR SMA DENGAN ALGORITMA AHP-NAÏVE  
BAYES**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:

Davin Ega Prasetya

NIM: 125150200111058



PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2016

## PENGESAHAN

SISTEM PENGKLASIFIKASI PENENTUAN JURUSAN KULIAH UNTUK PELAJAR SMA  
DENGAN ALGORITMA AHP-NAÏVE BAYES

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Davin Ega Prasetya

NIM: 125150200111058

Skripsi ini telah diuji dan dinyatakan lulus pada  
11 Agustus 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Imam Cholissodin, S.Si, M.Kom

NIK. 201201 850719 1 001

Agus Wahyu Widodo, S.T, M.Cs

NIP. 19740805 200112 1 001

Mengetahui

Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D

NIP. 19710518 200312 1 001

## PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 28 Juli 2016



Davin Ega Prasetya

NIM: 125150200111058

## KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat, tuntunan dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul **“Sistem Pengklasifikasi Penentuan Jurusan Kuliah Untuk Pelajar Sma Dengan Algoritma Ahp-Naïve Bayes”** sebagai salah satu persyaratan untuk menyelesaikan studi di Jurusan Informatika, Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.

Penulis menyadari bahwa tugas akhir ini dapat terselesaikan berkat bantuan, petunjuk, bimbingan dan dukungan dari berbagai pihak yang telah banyak membantu proses penyelesaian tugas akhir ini. Oleh karena itu penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada :

1. Imam Cholissodin, S.Si, M.Kom selaku pembimbing I dan Agus Wahyu Widodo, S.T, M.Cs selaku pembimbing II yang telah membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini.
2. Orang tua penulis, R.M Soetadi dan Ibu Wahyu Sulistyningtyas yang tak henti-hentinya memberikan dukungan moril dan materil.
3. Saudara penulis, Angga Setiawan dan Rio Rama Pradipta yang telah memberikan motivasi dan semangat demi terselesaikannya skripsi ini.
4. Seluruh dosen Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya atas kesediaan membagi ilmunya kepada penulis sehingga penulis dapat menyelesaikan skripsi ini.
5. Sahabat penulis di dalam kelompok tumbuh bersama Yuni Widyaningtyas, Leo Tiofan J Silalahi, Mariani, Luhur Dimas Putranto yang selalu menemani, bertukar ilmu, semangat dan motivasi kepada penulis selama masa studi dan penyusunan skripsi.
6. Adik rohani R.P Hero Wijaya, Harun Christian Bentro, dan Maria Sartika Tambunan yang selalu memberikan semangat, motivasi, doa, dan keceriaan di dalam hidup penulis.
7. Sahabat penulis Yunastria Christine Irwanti dan Canny Amerilyse Caesar yang telah berperan banyak dan membantu penulis di dalam penyelesaian skripsi ini.
8. Sahabat penulis yang di dalam kepengurusan PMK Daniel Komisi 4 Mahisa Dyan Dipta, Elisa J.I Siahaan, Tyrse Rezza Biantong, Randa Alverdian yang telah memberikan rasa kekeluargaan di dalam melayani
9. Teman-teman PMK Daniel angkatan 2012 yang selalu menghibur, memberikan motivasi dan semangat serta rasa kebersamaan kepada penulis selama ini.

10. Seluruh keluarga besar PMK Daniel terimakasih atas kebersamaan, keceriaan, saran dan dorongan kepada penulis selama masa studi.
11. Teman-teman seperjuangan Program Teknologi Informasi dan Ilmu Komputer angkatan 2012 yang telah memberikan bantuan selama masa studi hingga penyelesaian skripsi ini.
12. Dan semua pihak yang tidak bisa disebutkan satu per satu. Terima kasih atas segala bantuannya.

Penulis sadar bahwa skripsi ini masih banyak kekurangan, oleh karena itu kritik dan saran yang bersifat membangun sangat diharapkan untuk menyempurnakan skripsi ini. Penulis berharap skripsi ini dapat bermanfaat khususnya bagi diri sendiri dan bagi semua pihak.

Malang, 8 Agustus 2016

Penulis

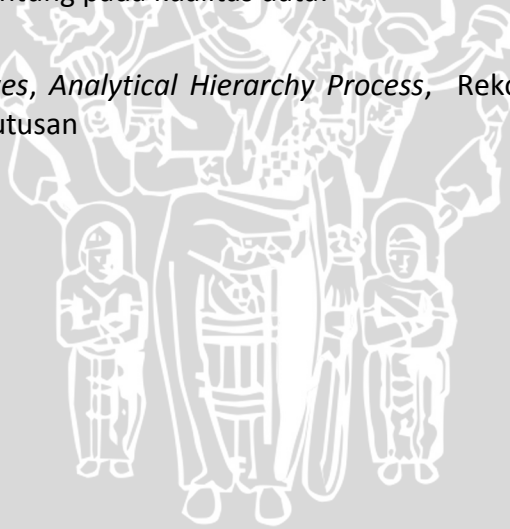


## ABSTRAK

Perkuliahan di universitas sudah menjadi kebutuhan bagi pelajar SMA di dalam meneruskan pendidikannya. Namun masih banyak pelajar SMA yang belum mengetahui jurusan yang tepat untuk mereka. Salah satu faktor permasalahan ini adalah kurangnya pengetahuan pelajar tersebut terhadap minat dan kemampuannya. Faktor lainnya yang juga turut berperan di permasalahan ini adalah tuntutan orang tua terhadap anaknya untuk mendaftar di suatu jurusan tertentu, padahal belum tentu jurusan tersebut sesuai dengan minat dan kemampuannya.

Ditinjau dari permasalahan tersebut, maka sistem klasifikasi jurusan ini dapat digunakan untuk membantu pelajar SMA di dalam memilih jurusan yang tepat berdasarkan kemampuan akademiknya. Metode Naïve Bayes dapat digunakan untuk mengklasifikasi data nilai pelajar ke dalam kelas fakultas. Selanjutnya metode Analytical Hierarchy Process digunakan untuk membantu menentukan jurusan yang tepat berdasarkan jurusan di fakultas yang telah ditentukan oleh metode Naïve Bayes. Dengan menggunakan metode AHP-Naïve Bayes ini dapat dihasilkan akurasi sistem yang mencapai 95% dengan catatan akurasi ini sangat bergantung pada kualitas data.

Kata Kunci : *Naïve Bayes*, *Analytical Hierarchy Process*, Rekomendasi Jurusan, Sistem Pendukung Keputusan

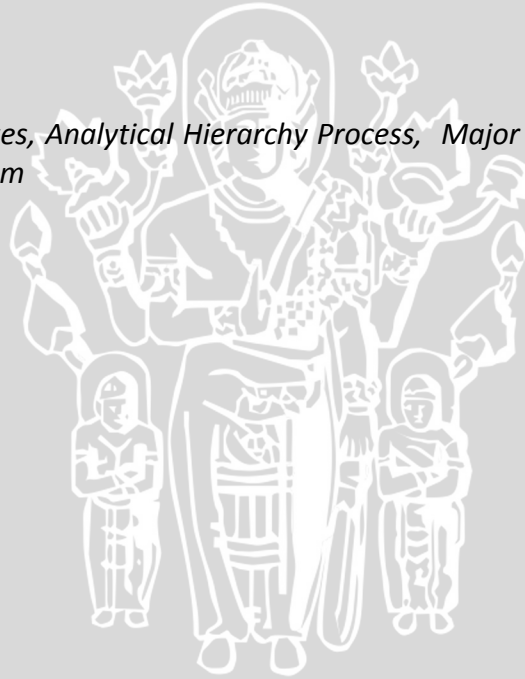


## ABSTRACT

*Studying at university has become a need for highschool student to continue their education. But there still many highschool student who not know college major that suit them. One of the factor is the student lack of knowledge about his own interest and capability. The other factor that has role in this problem is parents demand their child to enter to certain major, even though that major is not really suit with their interest and capability.*

*Judging from this problem, then major classification system can be used to help highschool student to choose the major that appropriate based on their academic capability. Naïve Bayes method can be used to classify student mark data to faculty class. Then Analytical Hierarchy Process method is used to help choosing the appropriate major based on major in faculty that Naïve Bayes method already determined. Using AHP-Naïve Bayes method, this system can generate accuracy which is up to 95 % with a note that this accuracy is very dependant to data quality.*

**Keywords :** *Naïve Bayes, Analytical Hierarchy Process, Major Recommendation, Decision Support System*



## DAFTAR ISI

SISTEM PENGKLASIFIKASI PENENTUAN JURUSAN KULIAH UNTUK PELAJAR SMA DENGAN ALGORITMA AHP-NAÏVE BAYES .....	i
PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT .....	vii
DAFTAR ISI.....	i
DAFTAR TABEL.....	iv
DAFTAR GAMBAR.....	vi
<b>BAB 1 PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan .....	2
1.4 Manfaat.....	2
1.5 Batasan masalah .....	3
1.6 Sistematika pembahasan.....	3
<b>BAB 2 LANDASAN KEPUSTAKAAN .....</b>	<b>5</b>
2.1 Kajian Pustaka .....	5
2.2 Dasar Teori.....	7
2.2.1 Keminatan Jurusan Kuliah.....	7
2.2.2 Klasifikasi.....	9
2.2.3 AHP.....	9
2.2.4 Naïve Bayes .....	12
<b>BAB 3 METODOLOGI .....</b>	<b>14</b>
3.1 Studi Literatur .....	14
3.2 Analisis Kebutuhan .....	14
3.3 Pengumpulan Data .....	15
3.4 Pengujian Algoritma.....	15
3.4.1 Pengujian Jumlah Data Latih Algoritma Naïve Bayes .....	16



3.4.2 Pengujian Jumlah Data Latih Algoritma Naïve Bayes dengan Data Latih Diskrit .....	16
3.4.3 Pengujian Konsistensi Hasil Sistem dengan Model AHP berbeda .....	16
3.5 Perancangan Sistem.....	17
3.6 Implementasi Sistem .....	17
3.7 Penarikan Kesimpulan .....	17
3.8 Penyelesaian Masalah Menggunakan <i>AHP-Naïve Bayes</i> .....	17
3.8.1 Penyelesaian dengan <i>Naïve Bayes</i> .....	19
3.8.1.1 Input Data Latih.....	20
3.8.1.2 Menghitung Prior dari Masing Masing Kelas .....	21
3.8.1.3 Input Data Uji .....	23
3.8.1.4 Menghitung Likelihood Data Uji .....	24
3.8.1.5 Menghitung Posterior dari Masing-masing Kemungkinan .....	28
3.8.2 Penyelesaian dengan <i>AHP</i> .....	31
3.8.2.1 Menentukan Bobot dan Persentase Nilai Mata Pelajaran .....	34
3.8.2.2 Uji Konsistensi Matriks.....	41
3.8.2.3 Menentukan Bobot dan Persentase Jurusan berdasarkan Nilai... ..	44
3.8.2.4 Menghitung Hasil Total .....	51
BAB 4 PERANCANGAN.....	54
4.1 Perancangan <i>User Interface</i> .....	54
4.1.1 Perancangan Halaman Utama .....	54
4.1.2 Perancangan Daftar Data .....	54
4.1.3 Perancangan <i>Setting</i> Data Uji .....	55
4.1.4 Perancangan <i>Testing Naïve Bayes</i> .....	55
4.1.5 Perancangan Manualisasi <i>Naïve Bayes</i> .....	56
4.1.6 Perancangan <i>Testing AHP</i> .....	56
4.1.7 Perancangan Manualisasi <i>AHP</i> .....	57
4.2 Perancangan Uji Coba dan Evaluasi.....	57
4.2.1 Uji Coba Jumlah Data Latih Algoritma <i>Naïve Bayes</i> .....	58
4.2.2 Uji Coba Jumlah Data Latih Algoritma Naïve Bayes dengan Data Latih Diskrit .....	58
4.2.3 Uji Coba Konsistensi Hasil Rekomendasi <i>AHP</i> .....	59



BAB 5 IMPLEMENTASI .....	61
5.1 Spesifikasi Sistem .....	61
5.1.1 Spesifikasi Perangkat Lunak .....	61
5.1.2 Spesifikasi Perangkat Keras .....	61
5.1.3 Batasan-Batasan Sistem .....	61
5.2 Implementasi Algoritma .....	62
5.2.1 Implementasi Algoritma <i>Naïve Bayes</i> .....	62
5.2.2 Implementasi Algoritma <i>AHP</i> .....	71
5.3 Implementasi <i>User Interface</i> .....	96
5.3.1 Implementasi Halaman Utama .....	96
5.3.2 Implementasi Halaman Daftar Data .....	97
5.3.3 Implementasi Halaman <i>Setting</i> Variabel .....	97
5.3.4 Implementasi Halaman <i>Testing Naïve Bayes</i> .....	98
5.3.5 Implementasi Halaman Manualisasi <i>Naïve Bayes</i> .....	98
5.3.6 Implementasi Halaman <i>Testing AHP</i> .....	99
5.3.7 Implementasi Halaman Manualisasi <i>AHP</i> .....	99
BAB 6 PENGUJIAN DAN ANALISIS .....	100
6.1 Hasil dan Analisa Jumlah Data Latih Algoritma <i>Naïve Bayes</i> .....	100
6.2 Hasil dan Analisa Jumlah Data Latih Algoritma <i>Naïve Bayes</i> dengan Data Latih Diskrit .....	101
6.3 Hasil dan Analisa Uji Coba Konsistensi Hasil Rekomendasi <i>AHP</i> .....	104
BAB 7 PENUTUP .....	106
7.1 Kesimpulan .....	106
7.2 Saran .....	106
DAFTAR PUSTAKA .....	108
LAMPIRAN 1 SURAT PERMOHONAN DATA SKRIPSI .....	109
LAMPIRAN 2 SURAT REKOMENDASI DINAS PENDIDIKAN .....	110
LAMPIRAN 3 DATA NILAI PELAJAR SMA .....	111
LAMPIRAN 4 PEMETAAN KEMINATAN JURUSAN KULIAH .....	135



## DAFTAR TABEL

Tabel 2.1 Kajian Pustaka .....	6
Tabel 2.2 Tingkat Kepentingan Perbandingan Kriteria .....	9
Tabel 2.3 Random Consistency Index .....	11
Tabel 3.1 Pengumpulan Data .....	15
Tabel 3.2 Data Latih Sampel.....	20
Tabel 3.3 Data Uji Sample .....	24
Tabel 3.4 Likelihood Fitur untuk Kemungkinan Diterima .....	27
Tabel 3.5 Likelihood Fitur untuk Kemungkinan Tidak.....	28
Tabel 3.6 Parameter Tiap Model AHP.....	32
Tabel 3.7 Aturan Pembobotan.....	34
Tabel 3.8 Data Uji Sampel .....	36
Tabel 3.9 Aturan Pembobotan.....	36
Tabel 3.10 Pembobotan Data Uji Sampel .....	36
Tabel 3.11 Keterangan Bobot .....	36
Tabel 3.12 Aturan Pembuatan Matriks.....	38
Tabel 3.13 Matriks Perbandingan Berpasangan .....	38
Tabel 3.14 Matriks Normalisasi Data .....	41
Tabel 3.15 Detail Persentase Mata Pelajaran .....	41
Tabel 3.16 Matriks Nilai Minimum Jurusan .....	45
Tabel 3.17 Matriks Pembobotan Jurusan .....	47
Tabel 3.18 Matriks Perbandingan Matematika .....	49
Tabel 3.19 Normalisasi Matriks Perbandingan Matematika .....	49
Tabel 3.20 Matriks Perbandingan Fisika .....	49
Tabel 3.21 Normalisasi Matriks Perbandingan Fisika .....	49
Tabel 3.22 Matriks Perbandingan Kimia .....	49
Tabel 3.23 Normalisasi Matriks Perbandingan Kimia .....	49
Tabel 3.24 Matriks Perbandingan Biologi .....	50
Tabel 3.25 Normalisasi Matriks Perbandingan Biologi .....	50
Tabel 3.26 Matriks Perbandingan TIK .....	50
Tabel 3.27 Normalisasi Matriks Perbandingan TIK .....	50

Tabel 3.28 Matriks Perbandingan Bahasa Indonesia .....	50
Tabel 3.29 Normalisasi Matriks Perbandingan Bahasa Indonesia .....	50
Tabel 3.30 Matriks Perbandingan Bahasa Inggris .....	51
Tabel 3.31 Normalisasi Matriks Perbandingan Bahasa Inggris .....	51
Tabel 3.32 Matriks Persentase Mata Pelajaran .....	51
Tabel 3.33 Matriks Persentase Total Mata Pelajaran .....	53
Tabel 4.1 Rancangan Uji Coba Jumlah Data Latih Algoritma Naïve Bayes .....	58
Tabel 4.2 Rancangan Uji Coba Jumlah Data Latih Algoritma Naïve Bayes dengan Data Latih Diskrit .....	58
Tabel 4.3 Rancangan Uji Coba Konsistensi Matriks Perbandingan Bobot Nilai Algoritma AHP .....	59
Tabel 4.4 Rancangan Uji Coba Konsistensi Hasil Rekomendasi AHP .....	59
Tabel 5.1 Spesifikasi Perangkat Lunak .....	61
Tabel 5.2 Spesifikasi Perangkat Keras .....	61
Tabel 6.1 Hasil Uji Coba Jumlah Data Latih Algoritma Naïve Bayes .....	100
Tabel 6.2 Hasil Uji Coba Jumlah Data Latih Algoritma Naïve Bayes Skenario 2..	101
Tabel 6.3 Keterangan Range Nilai .....	102
Tabel 6.4 Data yang Telah Diolah .....	102
Tabel 6.5 Hasil Uji Coba Jumlah Data Latih Algoritma Naïve Bayes dengan Data Latih Diskrit .....	103
Tabel 6.6 Daftar Data Uji Coba Konsistensi Hasil Rekomendasi AHP .....	104
Tabel 6.7 Hasil Uji Coba Konsistensi Hasil Rekomendasi AHP .....	104

## DAFTAR GAMBAR

Gambar 3.1 Diagram Blok Metodologi Penelitian .....	14
Gambar 3.2 Diagram Algoritma Penyelesaian Masalah .....	18
Gambar 3.3 Diagram Algoritma Naïve Bayes.....	19
Gambar 3.4 Daftar Kelas Hasil Klasifikasi Algoritma Naïve Bayes.....	20
Gambar 3.5 Diagram Perhitungan Prior Tiap Kelas .....	21
Gambar 3.6 Diagram Perhitungan Jumlah Data Tiap Kelas .....	22
Gambar 3.7 Diagram Perhitungan Prior Kelas .....	23
Gambar 3.8 Diagram Perhitungan Likelihood.....	24
Gambar 3.9 Menghitung Mean.....	25
Gambar 3.10 Menghitung Standar Deviasi .....	26
Gambar 3.11 Perhitungan Likelihood .....	27
Gambar 3.12 Diagram Perhitungan Posterior .....	29
Gambar 3.13 Diagram Perhitungan Nilai Posterior Tiap Kelas .....	30
Gambar 3.14 Diagram Perbandingan Nilai Posterior tiap Kelas .....	31
Gambar 3.15 Diagram Algoritma AHP .....	33
Gambar 3.16 Model Sistem Algoritma AHP.....	34
Gambar 3.17 Proses Menentukan Bobot dan Persentase Mata Pelajaran .....	35
Gambar 3.18 Pembuatan Matriks Perbandingan Berpasangan .....	37
Gambar 3.19 Penentuan Persentase Nilai Mata Pelajaran Calon Mahasiswa .....	39
Gambar 3.20 Pembuatan Matriks Normalisasi.....	40
Gambar 3.21 Proses Uji Konsistensi Matriks .....	42
Gambar 3.22 Proses Perhitungan $\lambda_{max}$ .....	43
Gambar 3.23 Proses Perhitungan CI .....	43
Gambar 3.24 Proses Perhitungan CR .....	44
Gambar 3.25 Proses Penentuan Bobot dan Persentase Jurusan .....	45
Gambar 3.26 Proses Perbandingan Nilai Calon Mahasiswa dengan Nilai Minimum Fakultas .....	47
Gambar 3.27 Proses Pembuatan Matriks Perbandingan Berpasangan dan Matriks Normalisasi.....	48
Gambar 3.28 Proses Menghitung Hasil Total .....	52
Gambar 4.1 Perancangan Halaman Utama.....	54

Gambar 4.2 Perancangan Halaman Daftar Data.....	55
Gambar 4.3 Perancangan Halaman Setting Data Uji .....	55
Gambar 4.4 Perancangan Halaman Testing Naïve Bayes .....	56
Gambar 4.5 Perancangan Halaman Manualisasi Naïve Bayes.....	56
Gambar 4.6 Perancangan Halaman Testing AHP .....	57
Gambar 4.7 Perancangan Halaman Manualisasi AHP .....	57
Gambar 5.1 Pendefinisian Data Calon Mahasiswa .....	62
Gambar 5.2 Pendefinisian Data Latih .....	65
Gambar 5.3 Perhitungan hasil Naïve Bayes .....	70
Gambar 5.4 Pengambilan Data Nilai Calon Mahasiswa .....	71
Gambar 5.5 Pengambilan Data Nilai Minimum Tiap Fakultas .....	73
Gambar 5.6 Pembuatan Matriks Nilai Calon Mahasiswa .....	76
Gambar 5.7 Pembobotan Mata Pelajaran .....	79
Gambar 5.8 Pembuatan Matriks Mata Pelajaran Model 1.....	94
Gambar 5.9 Perhitungan Persentase Akhir Jurusan .....	95
Gambar 5.10 Implementasi Halaman Utama .....	96
Gambar 5.11 Implementasi Halaman Daftar Data .....	97
Gambar 5.12 Implementasi Halaman Setting Variabel.....	97
Gambar 5.13 Implementasi Halaman Testing Naïve Bayes.....	98
Gambar 5.14 Implementasi Halaman Manualisasi Naïve Bayes.....	98
Gambar 5.15 Implementasi Halaman Testing AHP.....	99
Gambar 5.16 Implementasi Halaman Manualisasi AHP .....	99
Gambar 6.1 Grafik Hasil Akurasi Uji Coba Jumlah Data.....	101
Gambar 6.2 Grafik Hasil Akurasi Uji Coba Jumlah Data dengan Data Latih Diskrit .....	103



## BAB 1 PENDAHULUAN

### 1.1 Latar belakang

Pendidikan merupakan salah satu hal yang sangat penting, salah satunya adalah pendidikan di universitas. Bisa dikatakan penting karena pendidikan di universitas merupakan modal seseorang untuk meniti karier. Pendidikan yang tepat dapat membantu karier orang tersebut menuju kesuksesan di masa depan. Ketika seseorang mendapat pendidikan yang tepat, maka orang tersebut dapat mengeluarkan potensi yang maksimal di hidupnya, seperti bekerja, mendirikan sebuah usaha, dan lainnya.

Orang-orang yang menempuh suatu pendidikan tertentu di sekolah disebut dengan pelajar. Pelajar menempuh pendidikan supaya mendapat bekal pengetahuan di masa depannya. Rektor Institut Teknologi Surabaya Triyogi Yuwono di dalam seminar “Refleksi Tiga Tahun MP3EI” di Jakarta, Kamis (4/9) menyatakan jumlah partisipasi kasar penduduk negara Indonesia yang mengenyam pendidikan strata 1 baru 20 persen. Belum tentu seseorang dapat berkuliah di universitas tertentu dikarenakan banyak hal. Karena itu, persaingan seorang pelajar SMA untuk menikmati bangku kuliah sangat ketat, pelajar SMA pun dituntut untuk memilih strategi untuk dapat kuliah.

Salah satu strategi yang dapat dipilih mahasiswa adalah memilih jurusan yang sesuai dengan minat dan kemampuan, dikarenakan jika seorang pelajar memiliki minat dan kemampuan di suatu jurusan kuliah, maka segala tes yang disyaratkan oleh jurusan tersebut kemungkinan besar dapat dia lalui dengan baik. Namun, masih banyak pelajar, terutama pelajar sekolah menengah atas(SMA) yang masih bingung dalam menentukan pendidikan atau jurusan apa yang akan dia pilih di saat akan meneruskan pendidikannya ke universitas. Hal tersebut dikarenakan anak tersebut masih belum mengetahui minat dan kemampuannya. Selain itu, hal tersebut terjadi karena banyak pelajar yang ingin memilih jurusan yang berhubungan langsung dengan kariernya dikarenakan latar belakang pendidikan juga termasuk salah satu faktor pendukung kesuksesan seseorang ke depannya (Slim, 2014). Untuk kasus ini, terkadang orang tua berperan di dalam memilih jurusan anaknya. Orang tua terkadang menginginkan anaknya untuk mendaftar di satu jurusan tertentu karena memiliki prospek kerja yang bagus, padahal belum tentu pilihan jurusan dari orang tuanya sudah sesuai dengan minat dan kemampuan anaknya dan belum tentu 100 persen sesuai dengan pilihan di hati anaknya.

Jurusan kuliah yang tidak tepat bagi pelajar akan mengakibatkan pelajar tersebut akan terhambat perkembangan pendidikannya di perkuliahannya, dikarenakan ketidaknyamanan dengan materi-materi perkuliahan yang sebenarnya tidak seberapa dia sukai. Hal itu akan berdampak pada indeks prestasi (IP) anak tersebut yang bisa saja berada di bawah standar dikarenakan minat belajar yang tidak besar. Hal itu juga dapat berdampak pada kehidupan sosial anak tersebut. Karena merasa lingkungan perkuliahan yang telah dia jalani bukanlah

lingkungan dia yang sebenarnya, sehingga dia akan menutup diri terhadap orang-orang di sekitarnya.

Dalam penentuan suatu pilihan, dapat dilakukan secara matematis dengan beberapa metode. Pavani, Sharma, Hota(2013) membuat sebuah evaluasi berkelompok untuk guru dengan menggunakan metode AHP-TOPSIS. Sistem ini dibuat untuk menentukan pengajar dengan hasil evaluasi yang terbaik untuk mengajar di suatu kelas tertentu. Di kasus lain, Metode AHP-TOPSIS digunakan di dalam penelitian oleh Zhu dan Liu(2011) untuk mengevaluasi dan menentukan tingkat keamanan beberapa situs pemerintah. Cara evaluasi yang dilakukan adalah dengan menghitung beberapa variabel yang berkaitan dengan tingkat keamanan. Dengan menggunakan metode tersebut, situs pemerintah dapat diklasifikasikan ke beberapa tingkat kategori keamanan. Di dalam penelitian ini, akan dibahas mengenai Sistem Pengklasifikasi yang dapat membantu pelajar SMA untuk memilih jurusan yang tepat baginya berdasarkan beberapa faktor dan variabel yang dia punyai. Di dalam penelitian ini akan menggunakan metode AHP-Naïve Bayes. Metode AHP-Naïve Bayes telah digunakan Chan, Chung, Chow, Niu (2013) untuk membuat sebuah sistem yang dapat menentukan tingkat kepuasan kepada beberapa perusahaan distributor dengan menggunakan metode AHP-Naïve Bayes. Sistem yang dibangun memiliki keunggulan di dalam pembobotan kriteria di dalam sistem yang dapat dilakukan secara otomatis tanpa campur tangan ahli, sehingga proses penentuan hasil dapat berjalan secara cepat dan dinamis. Penelitian ini juga akan mencoba untuk mengklasifikasikan beberapa pelajar ke dalam beberapa kategori jurusan yang cocok untuk pelajar tersebut.

## 1.2 Rumusan masalah

1. Bagaimana menerapkan metode AHP-Naïve Bayes ke dalam Sistem Pengklasifikasi untuk penentuan jurusan kuliah yang tepat kepada pelajar SMA.
2. Bagaimana tingkat akurasi Sistem Pengklasifikasi untuk penentuan jurusan kuliah yang tepat kepada pelajar SMA.

## 1.3 Tujuan

1. Menerapkan metode AHP dan Naïve Bayes ke dalam Sistem Pengklasifikasi untuk penentuan jurusan kuliah yang tepat kepada pelajar SMA.
2. Menguji tingkat akurasi Sistem Pengklasifikasi untuk penentuan jurusan kuliah yang tepat kepada pelajar SMA dengan metode AHP dan Naïve Bayes.

## 1.4 Manfaat

Penulisan penelitian ini diharapkan mempunyai manfaat yang baik dan berguna bagi pembaca dan penulis. Adapun manfaat yang diharapkan adalah sebagai berikut:

1. Untuk Pelajar SMA :
  - Diharapkan dapat membantu memudahkan pelajar-pelajar SMA yang masih kebingungan di dalam menentukan jurusan kuliah yang tepat dengan dirinya.



2. Untuk Penulis :
  - Dapat mengetahui tingkat akurasi terbaik pada metode AHP dan Algoritma Naïve Bayes untuk pengklasifikasian jurusan terbaik bagi pelajar SMA.
  - Dapat mengembangkan sistem klasifikasi terbaru ke depannya dengan metode yang sudah ada maupun dengan metode baru.
3. Untuk Sekolah Menengah Atas :
  - Dapat membantu pengajar di SMA untuk mengarahkan siswanya kepada satu keminatan kuliah yang tepat untuknya.
4. Untuk Universitas :
  - Dapat membantu Universitas mendapatkan calon mahasiswa yang tepat untuk setiap jurusan sehingga kedepannya mahasiswa tersebut dapat memberikan kemajuan untuk Universitasnya.

### 1.5 Batasan masalah

Agar permasalahan yang dirumuskan dapat lebih terfokus dan tidak meluas, maka batasan-batasan yang ditentukan pada penelitian ini yaitu:

1. Data latih dan data uji yang digunakan dalam penelitian ini berasal dari data mata pelajaran yang berasal dari siswa SMAN 1 Malang.
2. Penentuan rekomendasi jurusan kuliah hanya diproses berdasarkan indeks nilai rapor siswa.
3. Proses perhitungan menggunakan metode AHP-Naïve Bayes dilakukan secara terpisah. Metode Naïve Bayes digunakan untuk menentukan fakultas dan metode AHP digunakan untuk menentukan jurusan.

### 1.6 Sistematika pembahasan

Sistematika penulisan yang digunakan pada tugas akhir ini sebagai berikut :

- |                |  |
|----------------|--|
| <b>BAB I</b>   | <b>PENDAHULUAN</b><br>Memuat latar belakang penelitian, rumusan masalah, batasan masalah, tujuan dan manfaat penulisan.  |
| <b>BAB II</b>  | <b>LANDASAN KEPUSTAKAAN</b><br>Bab ini membahas mengenai teori-teori yang berkaitan dan menunjang dalam penyelesaian tugas akhir ini.  |
| <b>BAB III</b> | <b>METODOLOGI</b><br>Bab ini membahas metode-metode yang digunakan perancangan, pengujian dan analisis dalam sistem pendukung keputusan untuk penentuan jurusan kuliah yang tepat. |
| <b>BAB IV</b>  | <b>PERANCANGAN</b><br>Bab ini membahas perancangan dari program yang telah dibuat.   |
| <b>BAB V</b>   | <b>IMPLEMENTASI</b><br>Bab ini membahas implementasi dari algoritma AHP-TOPSIS untuk penentuan jurusan kuliah yang tepat.  |

## **BAB VI**

### **PENGUJIAN DAN ANALISA**

Bab ini membahas tentang pengujian terhadap program yang telah dibuat dan dilakukan analisa terhadap program tersebut.

## **BAB VII**

### **KESIMPULAN DAN SARAN**

Dalam bab ini diberikan kesimpulan dan juga saran dari hasil penelitian berdasarkan uraian-uraian dari bab-bab sebelumnya, untuk pengembangan penelitian ini selanjutnya.



## BAB 2 LANDASAN KEPUSTAKAAN

Pada bab ini akan membahas tentang teori-teori yang menunjang penelitian, yaitu mengenai penjurusan keminatan, klasifikasi, algoritma AHP, dan algoritma Naïve Bayes.

### 2.1 Kajian Pustaka

Telah banyak dilakukan penelitian yang menggunakan metode klasifikasi di dalamnya. Pavani, Kumar, dan Hota menggunakan metode FAHP-TOPSIS untuk mengevaluasi kinerja dari guru di sebuah lembaga. Kinerja guru dievaluasi dengan memberikan bobot penilaian kepada beberapa kriteria yang dipertimbangkan dengan menggunakan metode Fuzzy AHP. Hasil perhitungan akan diperingkat dengan menggunakan metode TOPSIS untuk mengetahui guru dengan nilai evaluasi terbaik. Algoritma Fuzzy AHP – TOPSIS juga digunakan untuk mengevaluasi penyedia bahan bangunan dan menentukan index evaluasinya. Evaluasi penyedia bahan bangunan sangat penting karena pendistribusian bahan bangunan mendapat perhatian yang tinggi (Shusheng, Qin, Min, 2011). Di dalam sistem tersebut, metode FAHP digunakan untuk menentukan bobot index evaluasi terhadap bahan bangunan dan metode TOPSIS digunakan untuk menentukan evaluasi secara menyeluruh.

Metode yang sama juga digunakan oleh Yudatama dan Sarno dengan menambahkan teori fuzzy di dalam metode TOPSIS. Sistem tersebut berguna untuk menentukan tingkat manajemen resiko untuk perusahaan teknologi informasi. Teknologi informasi sangat membutuhkan manajemen informasi di dalam perusahaan yang menggunakannya untuk mendukung tujuan dari perusahaan tersebut, seperti digunakannya sumber daya sebaik mungkin dan resiko teknologi informasi dimanagemen dengan baik. Teori Fuzzy digunakan karena penilaian manusia sering tidak jelas dan tidak dapat diprediksi dengan angka numerik yang tepat, sehingga diperlukan teori fuzzy untuk menilai konsep ketidakpastian dari manusia. Di dalam aplikasi tersebut Fuzzy AHP digunakan untuk menentukan bobot dari kriteria tertentu dan Fuzzy TOPSIS untuk memberi peringkat terhadap alternative terpilih.

Pendekatan yang berbeda digunakan oleh Chan, Chung, dan Niu di dalam penentuan supplier dengan menggunakan metode AHP dan Naïve Bayes. AHP sangat bagus untuk menentukan 2 rata-rata yang berbeda di dalam penentuan keputusan berkelompok untuk mendapatkan prioritas kelompok (Chan, Chung, Niu, 2013). Sementara itu, Naïve Bayes digunakan untuk melakukan klasifikasi secara otomatis karena klasifikasi didapatkan dari perulangan nilai data yang telah diinputkan, sehingga campur tangan manusia dapat diminimalisir. Berikut ini adalah tabel perbandingan masing-masing metode yang telah dijelaskan.

**Tabel 2.1 Kajian Pustaka**

No	Judul	Penulis	Perbandingan	
			Kajian Pustaka	Skripsi Penulis
1.	A Group Expert Evaluation for Teachers by Integrating Fuzzy AHP and TOPSIS Models	Sirigiri Pavani, Lokesh Kumar Sharma, Hota H.S	- Menggunakan metode AHP dan TOPSIS untuk mengevaluasi kinerja dari guru	- Menggunakan metode Naïve Bayes untuk menghitung probabilitas semua kemungkinan dan metode AHP untuk melakukan penentuan kelas dari data berdasarkan perhitungan goal
2.	Evaluation Maturity Index and Risk Management for IT Governance Using Fuzzy AHP and Fuzzy TOPSIS	Uky Yudatama, Rianarto Sarno	- Menggunakan metode AHP untuk mendapatkan menentukan bobot dari kriteria yang ditentukan dan TOPSIS untuk memberi ranking dari website yang tersedia	- Menggunakan metode Naïve Bayes untuk menghitung probabilitas semua kemungkinan dan metode AHP untuk melakukan penentuan kelas dari data berdasarkan perhitungan goal
3.	Evaluation of building Materials Logistics Provider Based on FAHP-TOPSIS	Sun Shunsheng, Zhang Qin, Zhang Min	- Menggunakan metode Fuzzy AHP – TOPSIS untuk membuat evaluasi tentang kemampuan penyedia bahan bangunan untuk beradaptasi terhadap persaingan dan	- Menggunakan metode Naïve Bayes untuk menghitung probabilitas semua kemungkinan dan metode AHP untuk melakukan penentuan kelas

			memuaskan kebutuhan	dari data berdasarkan perhitungan goal
4.	An Integration of AHP Approach and Bayes Classification Algorithm in Supplier Selection	F.T.S Chan, S.H. Chung, J.C.L. Chow, B.Niu	- Menggunakan metode Naïve Bayes untuk menghitung probabilitas dari kriteria dan metode AHP untuk menentukan distributor dengan tingkat kepuasan tertinggi	- Menggunakan metode Naïve Bayes untuk menghitung probabilitas semua kemungkinan dan metode AHP untuk melakukan penentuan kelas dari data berdasarkan perhitungan goal

Pada contoh di atas, digunakan berbagai sistem penentuan suatu pilihan dengan menggunakan berbagai metode. Untuk metode di atas, terdapat beberapa metode gabungan dengan menggunakan metode TOPSIS sebagai penentu keputusan dengan sistem rangking. Metode tersebut sudah memberikan solusi yang terbaik dari berbagai alternative, namun sistem tersebut kurang dinamis karena data sistem akan tetap, sehingga data tidak akan terbaharui secara otomatis. Karena itu, dipilihlah metode AHP-Naïve Bayes untuk melakukan klasifikasi data ke dalam kelas-kelasnya masing-masing sekaligus, sehingga dapat membuat data semakin dinamis karena hasil klasifikasi data yang baru segera dimasukkan ke dalam data yang akan dipakai untuk proses klasifikasi berikutnya. Metode AHP-Naïve Bayes telah teruji di dalam masalah penentuan tingkat kepuasan terhadap distributor. Karena itu, metode AHP-Naïve Bayes juga diharapkan dapat menyelesaikan penentuan keminatan kuliah untuk pelajar SMA.

## 2.2 Dasar Teori

### 2.2.1 Keminatan Jurusan Kuliah

Jurusan kuliah yang tepat akan menentukan kesuksesan seseorang, karena jika seseorang tidak menikmati apa yang dia pelajari, maka Pribadi orang tersebut dapat tertekan dan kemungkinan terburuknya adalah anak tersebut tidak akan berkembang di bidang tersebut. Untuk itu, dibutuhkan faktor psikologis anak tersebut. Banyak hal yang perlu diketahui dari seseorang anak dari dalam dirinya seperti hobinya, keminatannya, dan juga kemampuannya. Beberapa faktor tersebut dapat menentukan jurusan kuliah yang tepat untuk seseorang anak.

Di dalam menentukan keminatan jurusan kuliah, telah dilakukan penelitian oleh Deorah, Sridharan, dan Goel (2010) mengenai sistem pakar untuk membantu pelajar menentukan jurusan pendidikannya bernama Student Advisor Expert

System (SAES). Penelitian tersebut dilatarbelakangi keadaan pelajar yang sering membuat pilihan yang kurang tepat dan sesuai dengan minat karena kurangnya sistem petunjuk yang lengkap. Sebagian besar pelajar di India memilih jurusan berdasarkan kecenderungan di masyarakat, ketertarikan sesaat, dan juga tekanan oleh orang tua, daripada ketertarikan personal dan kemampuan. Selain itu, menurut Novak dan Weiss (1988), pelajar yang yakin dengan jurusan yang dia pilih akan merasa lebih menikmati dan memiliki prestasi yang lebih baik.

SAES merupakan sistem pakar dengan menggunakan kombinasi case based dan rule based analysis di dalam pengambilan keputusannya. Dalam pengambilan keputusannya, SAES akan memberikan rekomendasi kuat, aman, dan lemah untuk jurusan dengan tingkat kesuksesan yang besar. Tingkat kesuksesan rekomendasi mencapai 68%. Melihat angka tingkat kesuksesan di dalam memberi rekomendasi cukup tinggi, maka sistem klasifikasi tersebut dapat berguna untuk membantu pelajar pada umumnya untuk memilih jurusan kuliah yang tepat.

Untuk memilih jurusan kuliah yang tepat, diperlukan kecermatan di dalam memilih, karena banyaknya keminatan kuliah yang tersedia saat ini. Keminatan jurusan kuliah terbagi dalam beberapa fakultas, yang merupakan kumpulan dari beberapa jurusan dengan kesamaan disiplin ilmu. Berdasarkan daftar fakultas yang terdapat di Universitas Brawijaya, contoh keminatan kuliah yaitu :

1. Fakultas Hukum
2. Fakultas Ekonomi dan Bisnis
3. Fakultas Ilmu Administrasi
4. Fakultas Pertanian
5. Fakultas Peternakan
6. Fakultas Teknik
7. Fakultas Kedokteran
8. Fakultas Perikanan dan Ilmu Kelautan
9. Fakultas Matematika dan Ilmu Pengetahuan Alam
10. Fakultas Teknologi Pertanian
11. Fakultas Ilmu Sosial dan Ilmu Politik
12. Fakultas Ilmu Budaya
13. Fakultas Kedokteran Hewan
14. Fakultas Kedokteran Gigi
15. Fakultas Ilmu Komputer
16. Program Vokasi
17. Program Pascasarjana

Masing-masing keminatan memiliki kriteria yang harus dipenuhi oleh calon Mahasiswa / pelajar. Tiap keminatan juga memiliki standar yang berbeda-beda untuk diujikan kepada calon mahasiswa / pelajarnya. Ada keminatan yang tidak

perlu keahlian di bidang matematika, tetapi membutuhkan kemampuan di bidang pengetahuan umum, begitu pula sebaliknya. Karena begitu banyaknya keminatan kuliah, tiap calon mahasiswa / pelajar perlu mempelajari kemampuannya terhadap tiap kriteria yang dibutuhkan di masing-masing keminatan kuliah tersebut.

### 2.2.2 Klasifikasi

Klasifikasi merupakan suatu proses yang bertujuan untuk mengelompokkan beberapa data ke dalam beberapa kelas. Proses pengelompokan dapat menggunakan beberapa metode. Beberapa metode yang termasuk di dalam klasifikasi merupakan Bayesian Decision, Naïve Bayes dan Linear Discriminant Analysis.

### 2.2.3 AHP

AHP pertama kali diusulkan oleh T.L. Saaty pada 1970. Konsep dasar dari AHP adalah membagi permasalahan kompleks ke dalam faktor penyusun dan mengirim faktor ke dalam sebuah relasi grup, dan akhirnya menyusun struktur hirarki (Zhu, Liu, 2011). Melalui perbandingan, kita dapat menentukan tingkat kepentingan dari faktor di dalam hirarki, kemudian menggabungkan penilaian untuk menentukan bobot total dari faktor tersebut. AHP menggabungkan penilaian pribadi dengan cara logis dan dipengaruhi imajinasi, pengalaman, dan pengetahuan untuk menyusun hirarki dari suatu masalah berdasarkan logika, intuisi dan pengalaman (Saaty, 1994). AHP dapat digunakan untuk menyelesaikan penentuan keputusan dengan multi kriteria sehingga dapat digunakan secara luas di berbagai bidang, seperti pendidikan, social, ekonomi, dan lain-lain.

Tahap yang dilalui dalam menggunakan metode AHP :

1. Melakukan perbandingan pada masing-masing kriteria
  - Tiap kriteria akan diuji kepentingannya di dalam perhitungan, dengan memberikan tingkat kepentingan di dalam tabel perbandingan kriteria. Tabel pedoman untuk menentukan tingkat kepentingan masing-masing kriteria adalah :

**Tabel 2.2 Tingkat Kepentingan Perbandingan Kriteria**

Tingkat Kepentingan	Keterangan
1	Kedua elemen sama pentingnya
3	Elemen yang satu sedikit lebih penting daripada elemen lainnya
5	Elemen yang satu lebih penting daripada elemen lainnya
7	Satu elemen jelas lebih mutlak penting daripada elemen lainnya

9	Satu elemen mutlak penting daripada elemen lainnya
2,4,6,8	Nilai-nilai antara/ Nilai Tengah dua nilai pertimbangan yang berdekatan
Kebalikan	Nilai kebalikan, $A(i,j)=1/A(j,i)$ . Dimana A adalah matrik perbandingan berpasangan antar elemen baik kriteria, sub-kriteria maupun alternatif tujuan.

2. Melakukan Normalisasi data

- Rumus untuk melakukan normalisasi adalah

$$X_{i,j}' = \frac{X_{i,j}}{\sum_{j=1}^n X_{i,j}} \quad (2.1)$$

Keterangan :

$X_{i,j}$  = Matriks dengan indeks (i,j)

$X_{i,j}'$  = Hasil normalisasi matriks indeks(i,j)

$\sum_{j=1}^n X_{i,j}$  = Sigma penjumlahan matriks  $X_{i,j}$  dengan indeks j=1 hingga n (jumlah data)

- Kemudian lakukan perhitungan bobot kriteria

$$A = \frac{\sum_{i=1}^n X_{i,j}'}{n} \quad (2.2)$$

Keterangan :

A = Matrik bobot kriteria

$X_{i,j}$  = Matriks dengan indeks (i,j)

n = Jumlah variabel data

3. Cari Consistency Index dan Consistency Ratio

- $\lambda_{max} = \frac{\sum_{j=1}^n \lambda_j}{n} \quad (2.3)$

- $CI = \frac{(\lambda_{max}-n)}{(n-1)} \quad (2.4)$

- $CR = \frac{CI}{RI} \quad (2.5)$

Keterangan :

$\lambda_{max}$  = Rata-rata nilai konsistensi dari variabel

n = Jumlah variabel data

CI = Nilai Consistency Index

RI = Nilai Random Index dari tabel Random Consistency Index

CR = Nilai Consistency Ratio





Untuk nilai RI, ditentukan dengan tabel Random Consistency Index , berdasarkan banyak variabel. Adapun tabel Random Consistency Index adalah sebagai berikut :

**Tabel 2.3 Random Consistency Index**

n	Nilai RI
1	0
2	0
3	0.58
4	0.9
5	1.12
6	1.24
7	1
8	1.41
9	1.45
10	1.49
11	1.51
12	1.48
13	1.56
14	1.57
15	1.59

- Kemudian lakukan pengecekan hasil CR ,jika nilai CR < 0.1 , maka tabel dinyatakan konsisten
- 4. Buat Ranking Alternatif Base Kriteria untuk tiap kriteria yang ditentukan
  - Dengan melakukan normalisasi pada data di tiap kriteria tersebut dengan menggunakan rumus 2.1, kemudian mencari bobot alternative untuk tiap kriteria dengan rumus 2.2
- 5. Melakukan perangkingan terakhir untuk pengambilan keputusan
  - *Alternatif = Matriks tiap base kriteria × bobot kriteria*
  - Membandingkan hasil dari matriks hasil, alternative dengan nilai tertinggi merupakan pilihan terbaik.

### 2.2.4 Naïve Bayes

Naïve Bayes merupakan metode klasifikasi yang sederhana yang menggunakan perhitungan dengan peluang (Chan dan Chung, 2013). Perhitungan naïve bayes dilakukan dimana setiap atribut data bersifat saling lepas satu sama lain.

Proses untuk mendapatkan hasil klasifikasi dari sebuah data uji adalah dengan membandingkan hasil posterior untuk tiap kemungkinan kelas. Kelas dengan posterior tertinggi akan menjadi hasil klasifikasi. Langkah-langkah yang dilakukan untuk menguji suatu data uji adalah :

#### 1. Hitung Prior tiap kelas

i. Rumus :

$$P(C_1) = \frac{\text{Jumlah Kemunculan Data Kelas } C_1}{\text{Total Data Keseluruhan}} \quad (2.6)$$

Keterangan :

$P(C_1)$  = Nilai Prior untuk kelas C

#### 2. Hitung Likelihood untuk tiap kriteria

i. Data kontinu

Untuk menghitung Likelihood dengan tipe data kontinu digunakan densitas Gauss dengan rumus :

$$P(X_1|C_1) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(X_1-\mu)^2}{2\sigma^2}} \quad (2.7)$$

Keterangan :

$P(X_1|C_1)$  = Nilai Likelihood kondisi  $X_1$  untuk kelas  $C_1$

$\pi$  = Nilai PI (22/7 atau 3.141592653589793)

$e$  = Nilai Euler (2.718281828459045)

$X_1$  = Nilai Data Kontinu

$\mu$  = Nilai Mean dari data latih

$\sigma$  = Nilai Standar Deviasi dari data latih

Rumus Tambahan :

$$\mu = \frac{\sum_{j=1}^n X_j}{n} \quad (2.8)$$

$$\sigma = \sqrt{\frac{(\sum_{j=1}^n X_j - \mu)^2}{n-1}} \quad (2.9)$$

Keterangan :

- $\sum_{j=1}^n X_j$  = Sigma penjumlahan data  $x$  dengan indeks 1 hingga  $n$
- $\bar{x}$  = Nilai mean dari data  $x$
- $n$  = jumlah data

ii. Data Diskrit

$$P(X_1|C_1) = \frac{\text{Jumlah Kemunculan Kriteria } X_1}{\text{Total Data pada Kelas } C_1} \quad (2.10)$$

Keterangan :

- $P(X_1|C_1)$  = Peluang Kondisi  $X_1$  pada kelas  $C_1$

3. Tentukan Posterior

Untuk Menghitung posterior 1 kelas berdasarkan data uji menggunakan :

$$P(C_1|X_1, X_2 \dots X_n) = P(C_1) \prod_{j=1}^n P(X_j | C_1) \quad (2.11)$$

Keterangan :

- $P(C_1|X_1, X_2 \dots X_n)$  = nilai posterior data dengan variabel  $X_1$  hingga  $X_n$  untuk kelas  $C_1$
- $P(C_i)$  = nilai prior untuk kelas  $C_i$
- $\prod_{j=1}^n P(X_j | C_1)$  = Sigma Perkalian untuk Likelihood kondisi  $X$  dengan indeks 1 hingga  $n$  untuk kelas  $C_i$

Kemudian, untuk menghitung posterior dari data uji adalah dengan membandingkan nilai posterior tiap kelas. Kelas dengan nilai posterior tertinggi akan menjadi hasil klasifikasi kelas dari data uji. Detail persamaan adalah :

$$P(C|X_1, X_2 \dots X_n) = \arg \max P(C_i) \prod_{j=1}^n P(X_j | C_i) \quad (2.12)$$

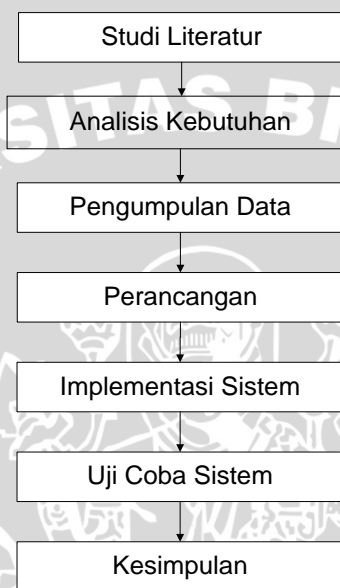
Keterangan :

- $\arg \max$  = argument dengan nilai tertinggi untuk hasil posterior tiap kelas
- $P(C) \prod_{j=1}^n P(X_j | C_i)$  = proses perhitungan posterior untuk kelas  $C_i$

Kelas dengan Posterior tertinggi akan menjadi hasil klasifikasi dari permasalahan. Metode Naïve Bayes dapat mengurangi biaya komputasi dengan performa yang baik dalam hal memprediksi dan klasifikasi (Manap, Tahir, Abdullah, 2012).

## BAB 3 METODOLOGI

Dalam bab metodologi ini membahas tentang metode yang digunakan untuk penelitian Sistem Pengklasifikasi Penentuan Jurusan Kuliah untuk Pelajar SMA dengan menggunakan metode AHP-Naïve Bayes. Tahapan penelitian ini meliputi studi literature, analisis kebutuhan, pengumpulan data, perancangan, implementasi sistem, uji coba sistem, dan kesimpulan. Langkah-langkah untuk penelitian ini dapat digambarkan di dalam Gambar 3.1



**Gambar 3.1 Diagram Blok Metodologi Penelitian**

### 3.1 Studi Literatur

Pada subbab ini membutuhkan studi literatur untuk membuat Sistem Pengklasifikasi Penentuan Jurusan Kuliah untuk Pelajar SMA dengan menggunakan metode AHP-Naïve Bayes. Pustaka dan informasi yang menyangkut sistem ini didapatkan dari beberapa situs di internet, dosen-dosen, rekan-rekan mahasiswa, dan buku. Teori yang dipelajari yaitu :

- Tentang Keminatan Jurusan Kuliah
- Klasifikasi
- AHP
- TOPSIS

### 3.2 Analisis Kebutuhan

Pada subbab ini terdiri dari dua jenis analisis kebutuhan, yaitu analisis kebutuhan fungsional dan analisis kebutuhan non fungsional. Sistem yang dibuat ini memiliki kebutuhan yaitu dapat menentukan keputusan di jurusan manakah seorang calon mahasiswa ditempatkan berdasarkan beberapa kriteria yang berdasarkan kemampuan akademik anak tersebut

Untuk kebutuhan ini berisi bagaimana sistem ini dapat menentukan jurusan yang cocok untuk seorang calon mahasiswa. Pengguna dapat menjalankan sistem tanpa harus login terlebih dahulu. Pengguna dapat menginputkan bobot dari masing-masing kriteria, melihat matriks bobot, melihat normalisasi dari matriks bobot, melihat persentase dari kriteria masing-masing berdasarkan kriteria yang dibutuhkan, kemudian dapat melihat hasil pemeringkatan dari beberapa jurusan kuliah yang dirasa cocok dengan calon mahasiswa tersebut.

### 3.3 Pengumpulan Data

Sistem ini untuk seluruh mahasiswa di Indonesia yang melakukan pendaftaran beasiswa. Variabelnya yaitu pada sistem pendukung keputusan ini apakah seorang mahasiswa layak untuk mendapatkan beasiswa atau tidak.

Untuk cara pengumpulan datanya menggunakan jenis kuisioner secara online. Pada Tabel 3.1 dapat digambarkan seperti apa data yang digunakan pada penelitian ini.

**Tabel 3.1 Pengumpulan Data**

No.	Kebutuhan Data	Sumber Data	Metode	Kegunaan Data
1.	Data akademik calon mahasiswa	SMAN 1 Malang	Langsung	Digunakan sebagai data untuk menentukan jurusan yang cocok untuk calon mahasiswa.
2.	Pengujian sistem pendukung keputusan untuk menentukan jurusan yang tepat untuk calon mahasiswa	Calon mahasiswa	Kuisioner	Untuk menentukan jurusan yang cocok untuk calon mahasiswa tersebut

### 3.4 Pengujian Algoritma

Pengujian dan pembahasan pada penelitian ini terdiri dari dua bagian, yaitu pengujian terkait hasil klasifikasi data menggunakan Naïve Bayes untuk mengetahui dan pengujian terkait hasil klasifikasi data menggunakan AHP untuk mengetahui jurusan. Skenario pengujian yang dilakukan antara lain sebagai berikut :

1. Pengujian jumlah data latih algoritma Naïve Bayes
2. Pengujian konsistensi matriks perbandingan bobot kriteria algoritma AHP
3. Pengujian akurasi sistem setelah update database

### 3.4.1 Pengujian Jumlah Data Latih Algoritma Naïve Bayes

Pengujian jumlah data latih algoritma naïve bayes bertujuan untuk mendapatkan jumlah data latih optimal sehingga didapatkan hasil klasifikasi dengan tingkat akurasi yang terbaik. Di dalam pengujian akan dilakukan pengujian dengan jumlah data latih yang berbeda, dengan skenario sebagai berikut :

1. Skenario pengujian dengan jumlah data latih sebanyak 90 persen dari jumlah data keseluruhan.
2. Skenario pengujian dengan jumlah data latih sebanyak 80 persen dari jumlah data keseluruhan.
3. Skenario pengujian dengan jumlah data latih sebanyak 70 persen dari jumlah data keseluruhan.
4. Skenario pengujian dengan jumlah data latih sebanyak 70 persen dari jumlah data keseluruhan.
5. Skenario pengujian dengan jumlah data latih sebanyak 60 persen dari jumlah data keseluruhan.
6. Skenario pengujian dengan jumlah data latih sebanyak 50 persen dari jumlah data keseluruhan.

### 3.4.2 Pengujian Jumlah Data Latih Algoritma Naïve Bayes dengan Data Latih Diskrit

Pengujian jumlah data latih algoritma naïve bayes dengan data latih diskrit bertujuan untuk mendapatkan jumlah data latih optimal sehingga didapatkan hasil klasifikasi dengan tingkat akurasi yang terbaik, dengan kondisi data latih telah diubah ke dalam bentuk data diskrit. Skenario pengujian adalah sebagai berikut :

1. Mengubah data latih menjadi data diskrit.
2. Skenario pengujian dengan jumlah data latih sebanyak 90 persen dari jumlah data keseluruhan.
3. Skenario pengujian dengan jumlah data latih sebanyak 80 persen dari jumlah data keseluruhan.

### 3.4.3 Pengujian Konsistensi Hasil Sistem dengan Model AHP berbeda

Pengujian konsistensi hasil sistem dengan model AHP berbeda bertujuan untuk mengetahui konsistensi rekomendasi jurusan yang diberikan dengan menggunakan 3 model AHP yang berbeda. Model AHP akan dijelaskan pada Tabel 3.6. Skenario pengujian adalah sebagai berikut :

1. Dilakukan pengujian akurasi dengan menggunakan model 1 AHP sebanyak 10 kali.
2. Dilakukan pengujian akurasi dengan menggunakan model 2 AHP sebanyak 10 kali.
3. Dilakukan pengujian akurasi dengan menggunakan model 3 AHP sebanyak 10 kali.
4. Dilakukan pencocokan hasil dari model 1, model 2, dan model 3 AHP.

### 3.5 Perancangan Sistem

Perancangan sistem dilakukan untuk mempermudah implementasi, pengujian serta analisis. Langkah – langkah yang dilakukan dalam perancangan sistem adalah sebagai berikut:

1. Perancangan Antarmuka Pengguna

Perancangan antarmuka pengguna dapat memudahkan pengguna menggunakan sistem yang akan dibangun

2. Perancangan Pengujian

Perancangan pengujian sistem yang dilakukan meliputi pengujian jumlah data latih algoritma Naïve Bayes, pengujian jumlah data uji algoritma Naïve Bayes, pengujian konsistensi matriks perbandingan bobot kriteria algoritma AHP

### 3.6 Implementasi Sistem

Implementasi sistem dibuat berdasarkan perancangan yang telah dibuat sebelumnya. Implementasi dilakukan dengan bahasa pemrograman JAVA menggunakan *software* Netbeans 8.0, database menggunakan MySQL, dan beberapa *tools* pendukung lainnya. Implementasi sistem pengklasifikasi penentuan jurusan kuliah untuk pelajar sma dengan algoritma ahp-naïve bayes meliputi :

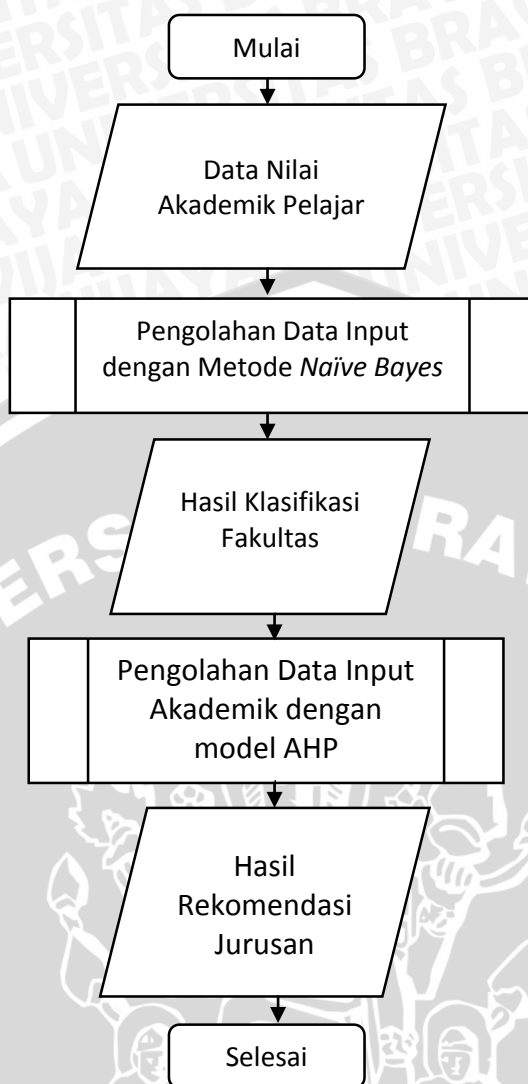
1. Penerapan algoritma Naïve Bayes dan AHP dalam program menggunakan bahasa pemrograman JAVA.
2. Pembuatan antarmuka program.
3. Penambahan data yang diperlukan ke dalam database MySQL

### 3.7 Penarikan Kesimpulan

Melakukan evaluasi dan analisis terhadap hasil pengujian perangkat lunak, kemudian ditarik ke dalam sebuah kesimpulan. Kesimpulan diambil untuk menjawab rumusan masalah yang telah ditetapkan sebelumnya. Tahap terakhir dari penulisan adalah saran yang berkenaan dengan hasil yang telah dicapai untuk memperbaiki kesalahan-kesalahan yang terjadi serta untuk memberikan pertimbangan untuk pengembangan penelitian lebih lanjut.

### 3.8 Penyelesaian Masalah Menggunakan *AHP-Naïve Bayes*

Pemberian rekomendasi jurusan kuliah dilakukan dengan mengolah data latih ke dalam metode *Naïve Bayes* untuk mendapatkan rekomendasi fakultas yang kemudian akan diolah dengan menggunakan metode *AHP* untuk mendapatkan rekomendasi jurusan kuliah yang tepat. Rancangan *flowchart* untuk system ini dapat dilihat pada Gambar 3.2 :



**Gambar 3.2 Diagram Algoritma Penyelesaian Masalah**

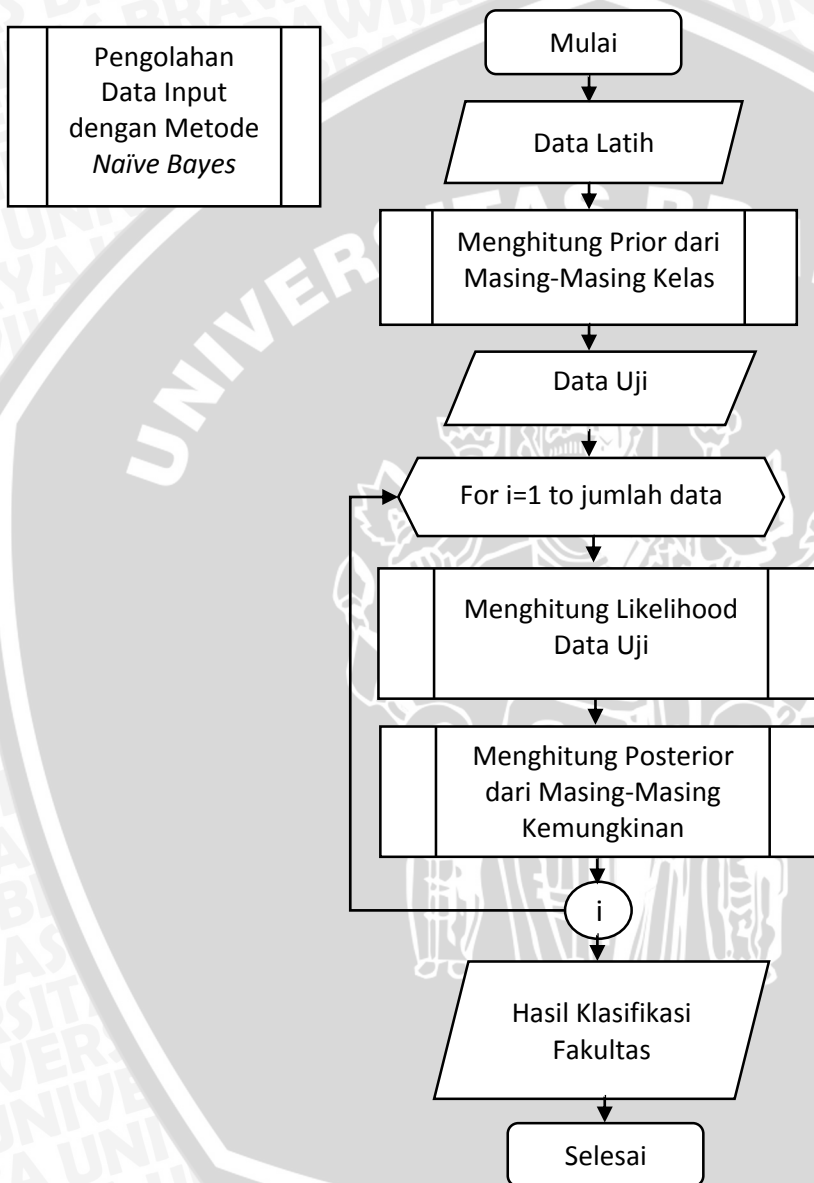
Langkah-langkah proses algoritma penyelesaian masalah berdasarkan Gambar 3.2 adalah sebagai berikut :

1. Sistem mendapatkan input data nilai pelajar yang telah diisi pada file dataTesting.txt
2. Sistem mengolah data input dengan menggunakan algoritma *naïve bayes* untuk mendapatkan hasil klasifikasi fakultas dari data input.
3. Sistem menampilkan output hasil klasifikasi fakultas dan meneruskan perhitungan data uji yang memiliki hasil diterima di suatu fakultas.
4. Sistem mengolah data uji yang telah diklasifikasikan ke fakultas dengan menggunakan algoritma *AHP* untuk mendapatkan rekomendasi jurusan.
5. Sistem menampilkan output berupa rekomendasi jurusan untuk tiap data uji yang telah diinputkan dan memiliki hasil klasifikasi diterima.

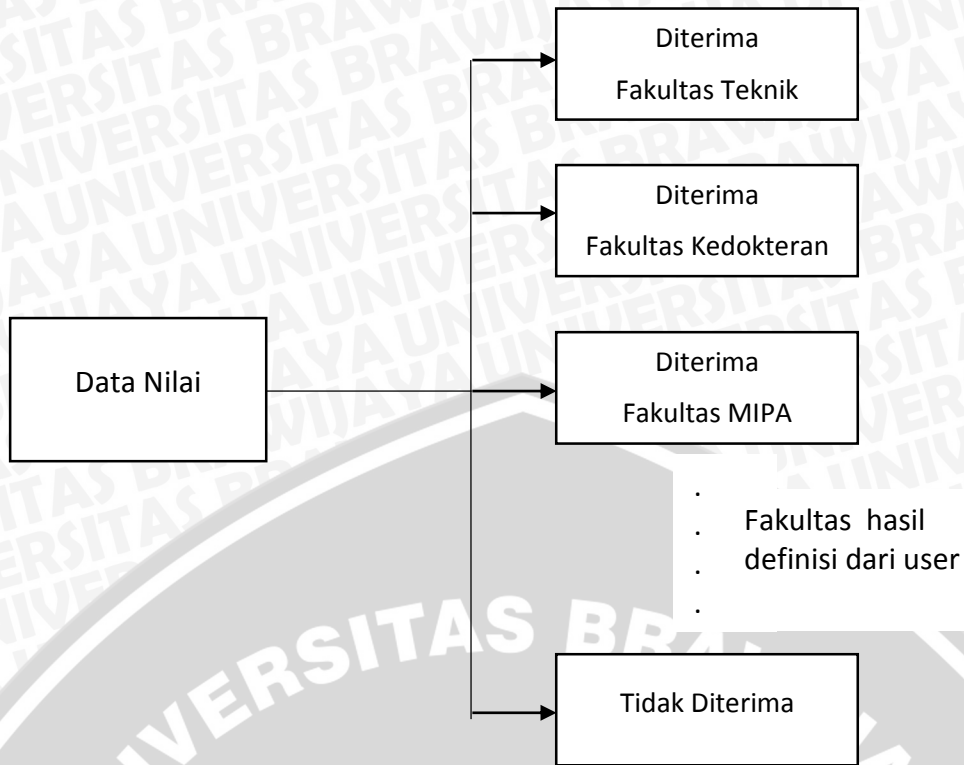


### 3.8.1 Penyelesaian dengan *Naïve Bayes*

Metode *Naïve Bayes* terdiri dari perhitungan prior  $P$  untuk setiap fitur, untuk data bertipe diskrit, prior didapatkan dari perulangan nilai di dalam atribut data. Untuk mendapatkan hasil dari klasifikasi, tiap prior dari fitur dikalikan sehingga didapatkan nilai posterior dari tiap kemungkinan yang ada. Alur proses metode *Naïve Bayes* dapat dilihat pada Gambar 3.3 dan proses klasifikasi metode *Naïve Bayes* dapat dilihat pada Gambar 3.4.



Gambar 3.3 Diagram Algoritma *Naïve Bayes*



**Gambar 3.4 Daftar Kelas Hasil Klasifikasi Algoritma Naïve Bayes**

Langkah-langkah proses algoritma naïve bayes berdasarkan Gambar 3.3 adalah sebagai berikut :

**3.8.1.1 Input Data Latih**

Data latih yang akan digunakan pada perhitungan memiliki fitur berupa nilai akademik dari calon mahasiswa. Pada bagian ini, digunakan data *sample* berupa data akademik dari calon mahasiswa berjumlah 10 orang. Data dapat dilihat di Tabel 3.2

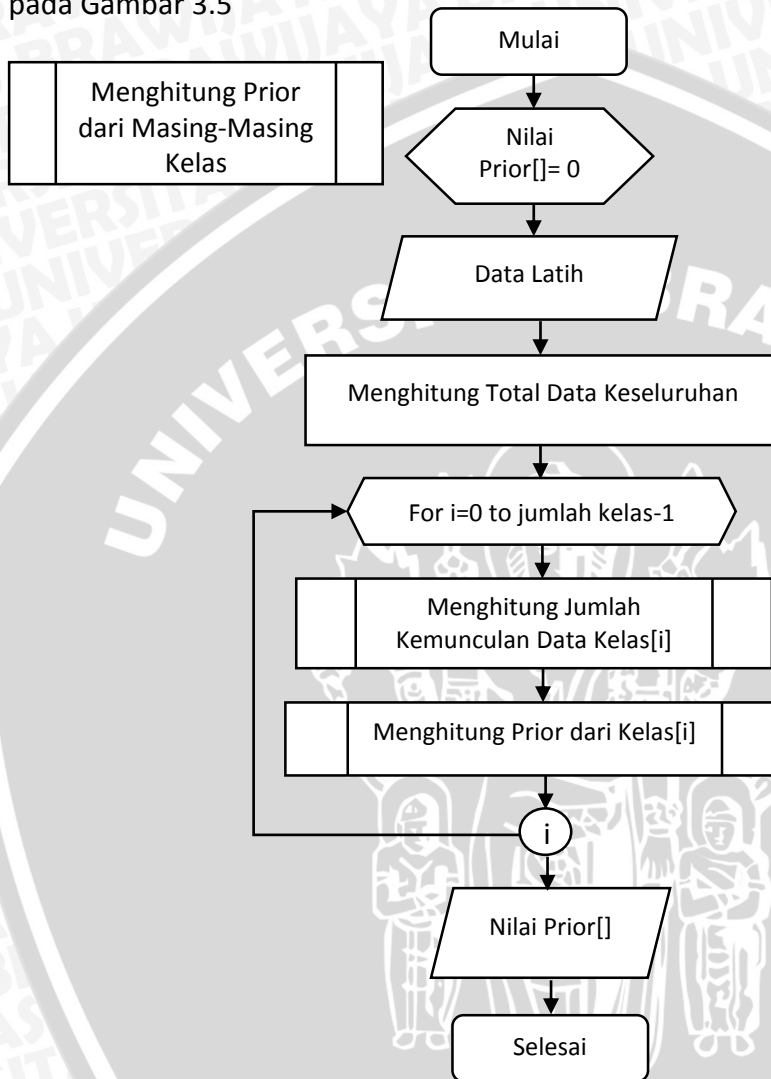
**Tabel 3.2 Data Latih Sampel**

Matematika	Fisika	Kimia	Biologi	TIK	Bind	Bing	Hasil
92,6667	90,3333	87,3333	88,3333	87,6667	89	85,3333	Diterima
82	81	82	86,6667	84,6667	86,6667	81	Diterima
86,3333	83,6667	80,3333	85,3333	86,6667	85,6667	86	Diterima
87,6667	81,6667	83,3333	89	86,3333	85,6667	83,3333	Diterima
95,3333	89,6667	903333	916667	85	84	82	Diterima
83,3333	81,3333	81	87	84	84,3333	81	Tidak
93,6667	82	81	88	83,3333	85,6667	85	Tidak
89	82	86,6667	86,3333	83,6667	84	83	Tidak
93,6667	84	81,3333	88,6667	88,3333	85,6667	85	Tidak
92,3333	86	82,6667	86,6667	86,6667	83,6667	81,3333	Tidak



### 3.8.1.2 Menghitung Prior dari Masing Masing Kelas

Di dalam naïve bayes, perhitungan prior data latih dapat dilambangkan dengan  $P(\text{kondisi})$ . Dalam kasus ini, terdapat 2 kemungkinan, yaitu diterima atau tidak, sehingga terdapat 2 prior yaitu  $P(\text{Diterima})$  dan  $P(\text{Tidak})$ . Untuk perhitungannya, digunakan persamaan 2.6 dengan alur proses yang dapat dilihat pada Gambar 3.5



**Gambar 3.5 Diagram Perhitungan Prior Tiap Kelas**

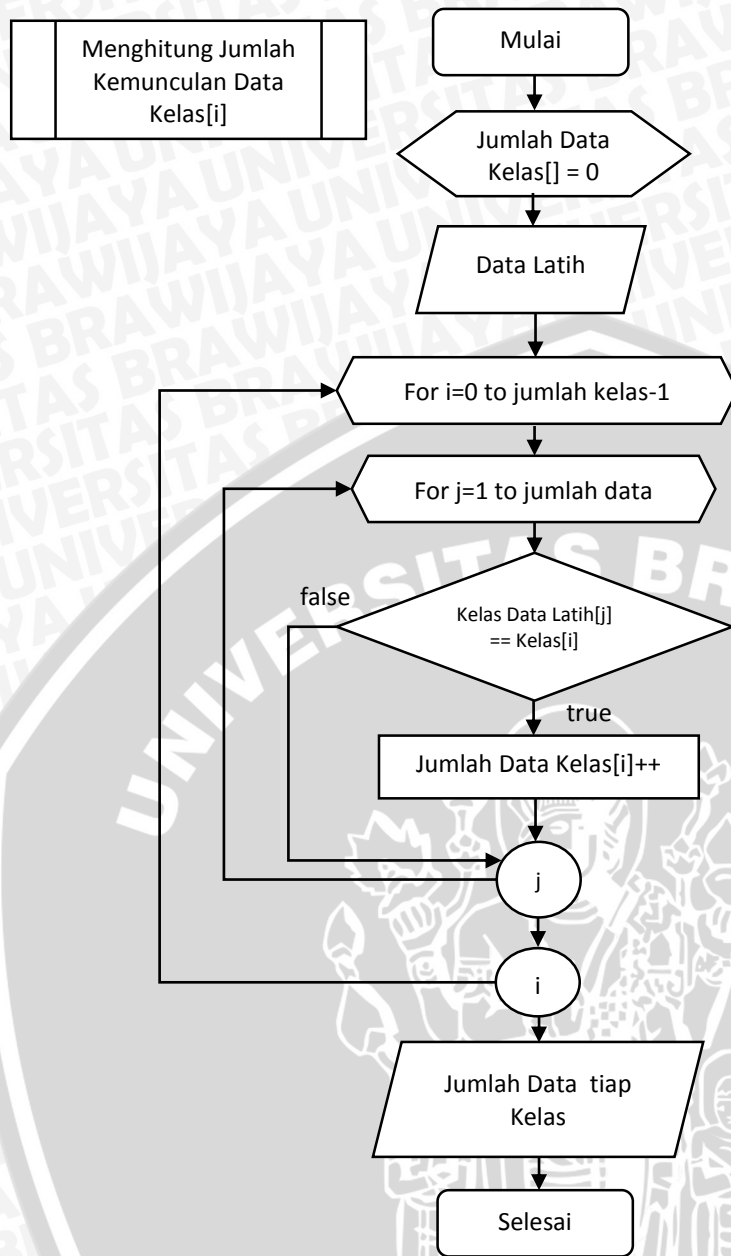
Penjelasan dari alur proses tersebut adalah :

a. Menghitung Total Data Keseluruhan

Data total yang dihitung adalah keseluruhan jumlah dari data latih. Untuk perhitungan kali ini digunakan data latih sample, dengan jumlah 10

b. Menghitung Jumlah Kemunculan Data Kelas

Data tiap kelas di dalam data latih dihitung, dengan menggunakan alur proses yang dijelaskan pada Gambar 3.6 :



**Gambar 3.6 Diagram Perhitungan Jumlah Data Tiap Kelas**

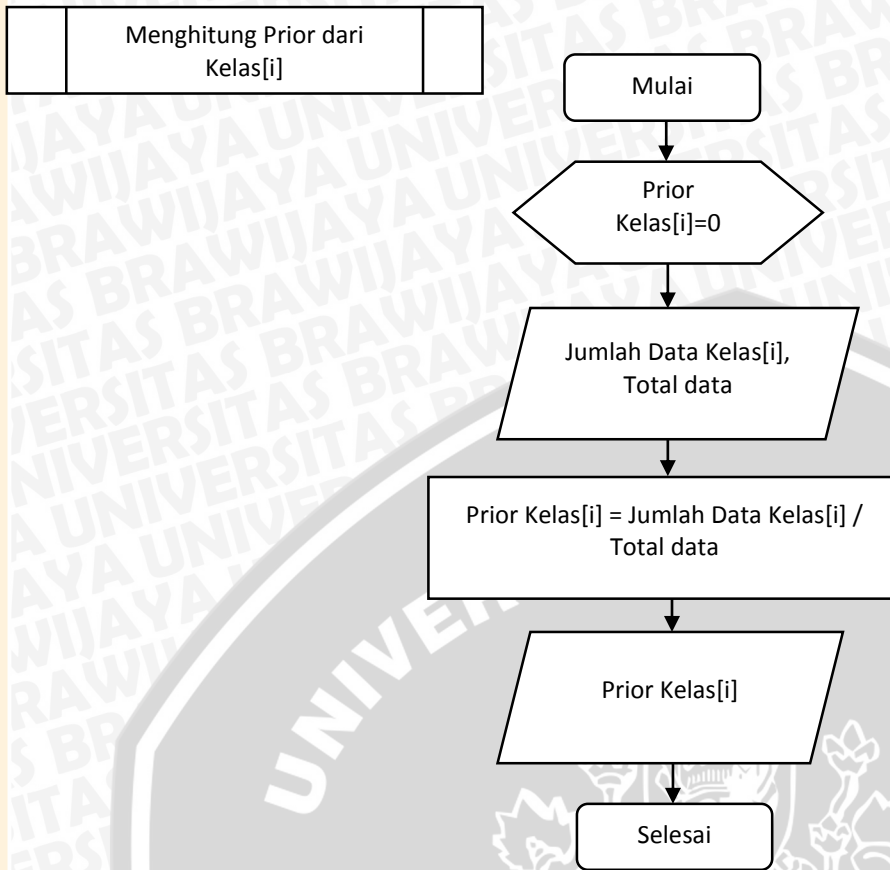
- Jumlah Data Kelas Diterima = 5
- Jumlah Data Kelas Tidak = 5

c. Menghitung Prior dari Kelas[i]

Menghitung prior dilakukan dengan persamaan 2.6 dengan hasil sebagai berikut :

$$\text{Persamaan : } P(C_1) = \frac{\text{Jumlah Kemunculan Data Kelas } C_1}{\text{Total Data Keseluruhan}}$$

Untuk diagram alur proses dapat dilihat pada Gambar 3.7



**Gambar 3.7 Diagram Perhitungan Prior Kelas**

Untuk perhitungan prior pada data latih sample, terdapat 2 kelas yaitu kelas diterima dan tidak, dengan detail perhitungan sebagai berikut :

- Prior Kelas Diterima

$$P(Diterima) = \frac{\text{Jumlah Kemunculan Data Kelas Diterima}}{\text{Total Data Keseluruhan}}$$

$$P(Diterima) = \frac{5}{10}$$

$$P(Diterima) = 0,5$$

- Prior Kelas Tidak

$$P(Tidak) = \frac{\text{Jumlah Kemunculan Data Kelas Tidak}}{\text{Total Data Keseluruhan}}$$

$$P(Tidak) = \frac{5}{10}$$

$$P(Tidak) = 0,5$$

### 3.8.1.3 Input Data Uji

Data uji yang akan digunakan memiliki atribut berupa nilai akademik dari calon mahasiswa. Data uji yang berada di bagian ini merupakan data uji *sample*, data uji dapat dilihat pada Tabel 3.3

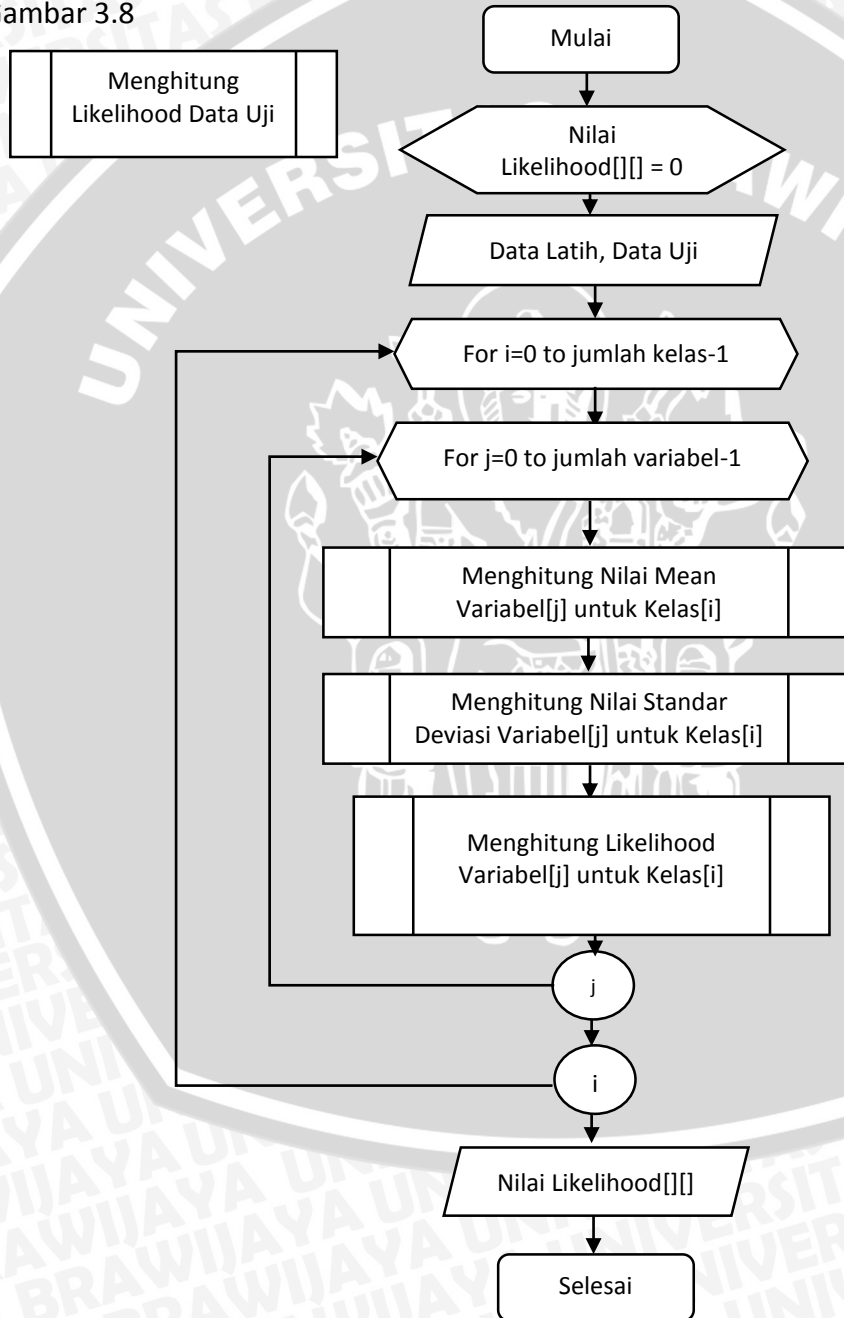


Tabel 3.3 Data Uji Sample

Matematika	Fisika	Kimia	Biologi	TIK	Bind	Bing
80	89	88	85	86	85	87

### 3.8.1.4 Menghitung Likelihood Data Uji

Untuk menghitung likelihood dari data uji, digunakan persamaan 2.7 dikarenakan data bertipe kontinu. Perhitungan likelihood dilakukan untuk masing masing fitur. Penjelasan metode perhitungan likelihood dapat dilihat pada Gambar 3.8

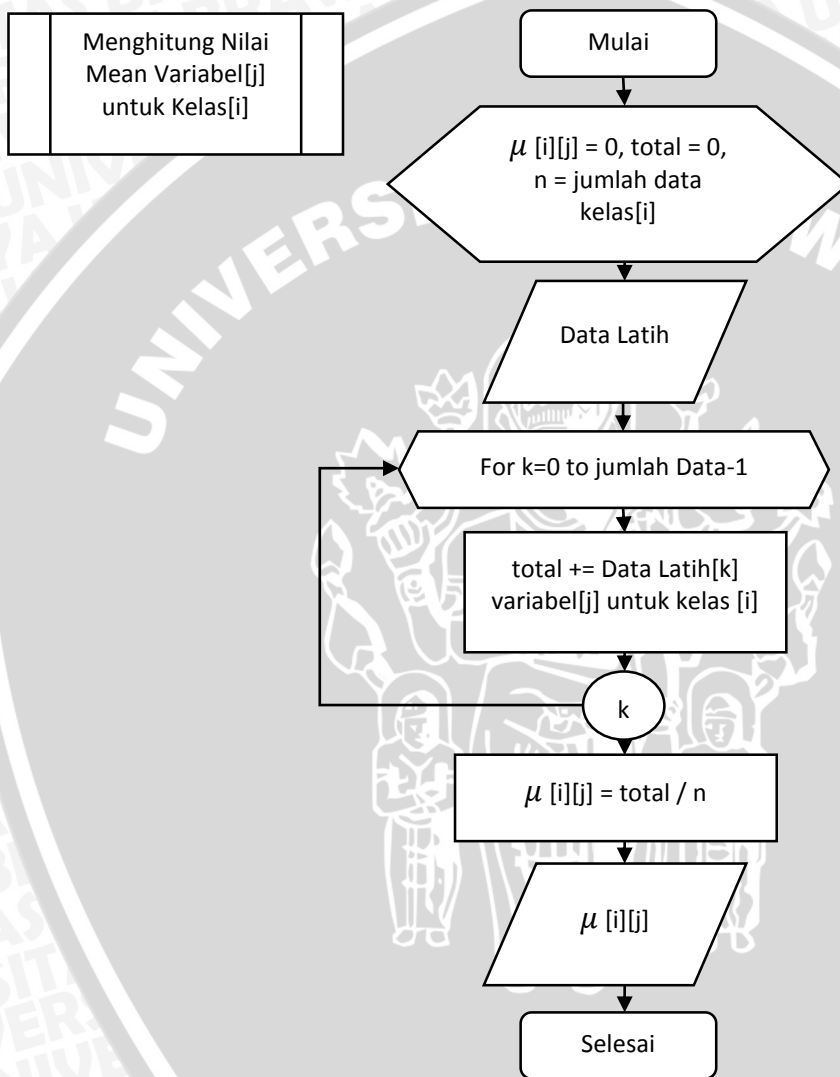


Gambar 3.8 Diagram Perhitungan Likelihood

Untuk penjelasan alur proses, digunakan contoh perhitungan untuk menghitung  $P(X=\text{Nilai Matematika} | \text{Diterima})$  yang memiliki perhitungan sebagai berikut :

- a. Menghitung Nilai Mean Variabel[j] untuk Kelas[i]

Alur proses untuk menghitung nilai mean variabel[j] kelas[i] dapat dilihat pada Gambar 3.9 :



**Gambar 3.9 Menghitung Mean**

Perhitungan mean ini menggunakan persamaan 2.8, dengan contoh perhitungan sebagai berikut :

(dengan variabel = matematika dan kelas = diterima)

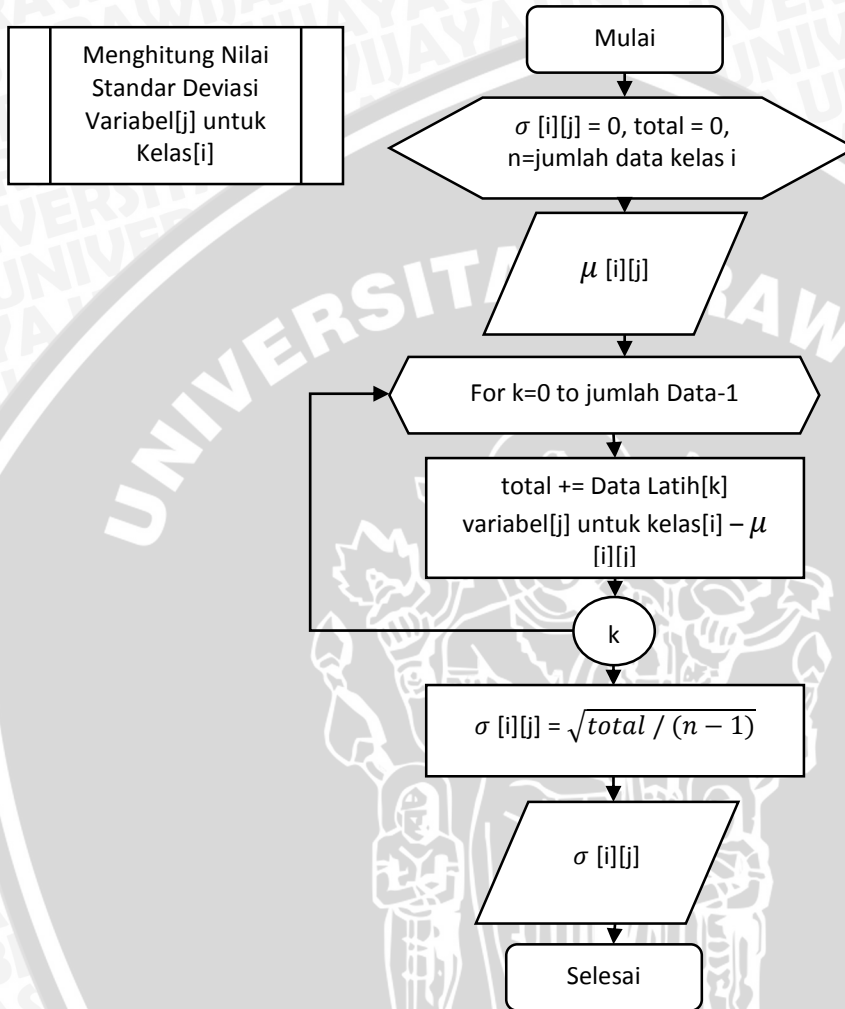
$$\mu = \frac{\sum_{j=1}^n X_j}{n}$$

$$\mu = (\text{Nilai Matematika Data 1} + \text{Nilai Matematika Data 2} + \dots + \text{Nilai Matematika Data } n) / 5$$

$$\mu = (92,6667 + 82 + 86,3333 + 87,6667 + 95,3333) / 5 = 88,8$$

b. Menghitung Nilai Standar Deviasi Variabel[j] untuk Kelas[i]

Alur Proses untuk menghitung nilai standar deviasi variabel[j] untuk kelas[i] dapat dilihat pada Gambar 3.10



Gambar 3.10 Menghitung Standar Devisasi

Perhitungan standar deviasi menggunakan persamaan 2.9, dengan contoh perhitungan sebagai berikut :

(dengan variabel = matematika dan kelas = diterima)

$$\sigma = \sqrt{\frac{\sum_{j=1}^n X_j - \bar{X}}{n - 1}}$$

$$\sigma = \sqrt{(Data\ Mat\ 1 - mean\ mat)^2 + (Data\ Mat\ 2 - mean\ mat)^2 \dots + (Data\ Mat\ n - mean\ mat)^2}$$

$$\sigma = \sqrt{(92,6667 - 89,6)^2 + (82 - 89,6)^2 + (86,3333 - 89,6)^2 + (87,6667 - 89,6)^2 + (95,3333 - 89,6)^2} / 5$$

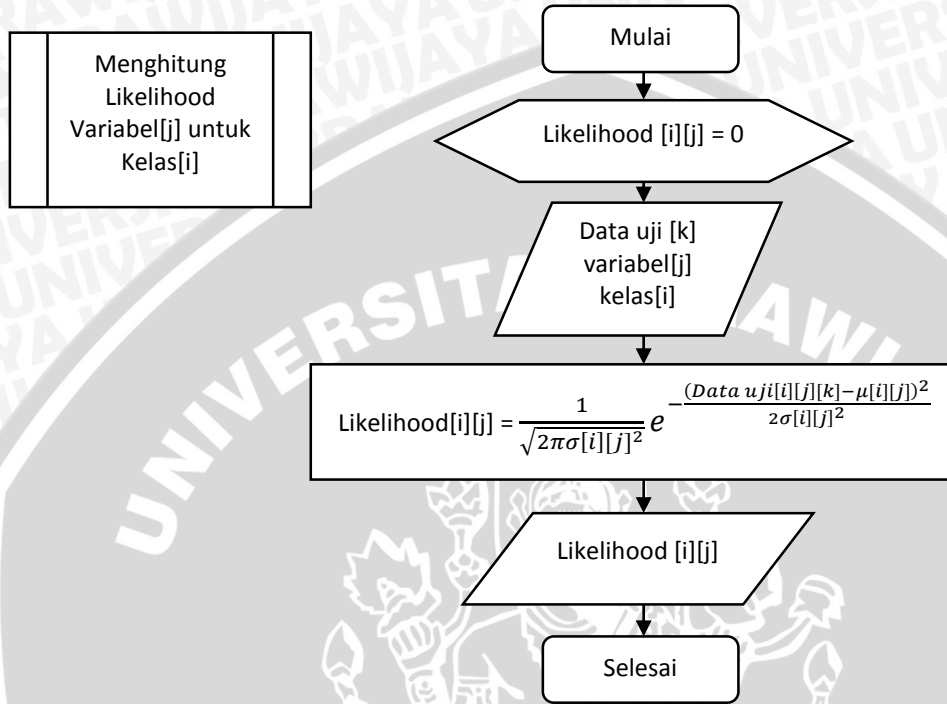




-  $\sigma = 5,27362199$

c. Menghitung Likelihood untuk Variabel [j] Kelas[i]

Alur proses untuk menghitung likelihood variabel[j] kelas[i] dapat dilihat pada Gambar 3.11



**Gambar 3.11 Perhitungan Likelihood**

Untuk menghitung likelihood menggunakan persamaan 2.7 dengan contoh perhitungan sebagai berikut :  
(dengan data = 80 dan kelas = diterima)

-  $P(X_1|C_1) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(X_1-\mu)^2}{2\sigma^2}}$

-  $P(80|Diterima) = \frac{1}{\sqrt{2\pi \cdot 5,27362199^2}} e^{-\frac{(80-88,8)^2}{2 \cdot 5,27362199^2}}$

-  $P(80|Diterima) = 0,018799851$

Hasil perhitungan likelihood masing-masing fitur dalam kelas secara lengkap dapat dilihat pada Tabel 3.4 dan Tabel 3.5

- Likelihood diterima :

**Tabel 3.4 Likelihood Fitur untuk Kemungkinan Diterima**

Likelihood	Persentase
P(Matematika=80   Diterima)	0,018799851
P(Fisika=89   Diterima)	0,063107677



Likelihood	Persentase
$P(\text{Kimia}=88   \text{Diterima})$	0,069980505
$P(\text{Biologi}=85   \text{Diterima})$	0,068570691
$P(\text{TIK}=86   \text{Diterima})$	0,322875813
$P(\text{Bind}=85   \text{Diterima})$	0,175561167
$P(\text{Bing}=87   \text{Diterima})$	0,049778479

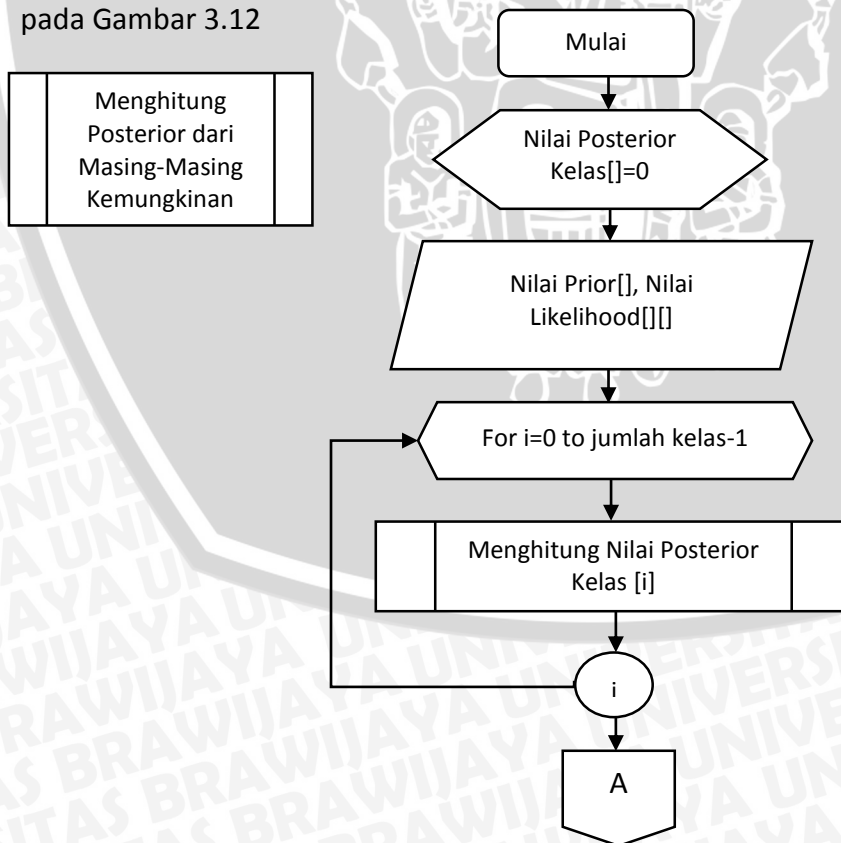
- Likelihood tidak diterima :

**Tabel 3.5 Likelihood Fitur untuk Kemungkinan Tidak**

Likelihood	Persentase
$P(\text{Matematika}=80   \text{Tidak})$	0,005474315
$P(\text{Fisika}=89   \text{Tidak})$	0,001758551
$P(\text{Kimia}=88   \text{Tidak})$	0,012649912
$P(\text{Biologi}=85   \text{Tidak})$	0,022990857
$P(\text{TIK}=86   \text{Tidak})$	0,170177501
$P(\text{Bind}=85   \text{Tidak})$	0,397502938
$P(\text{Bing}=87   \text{Tidak})$	0,025512577

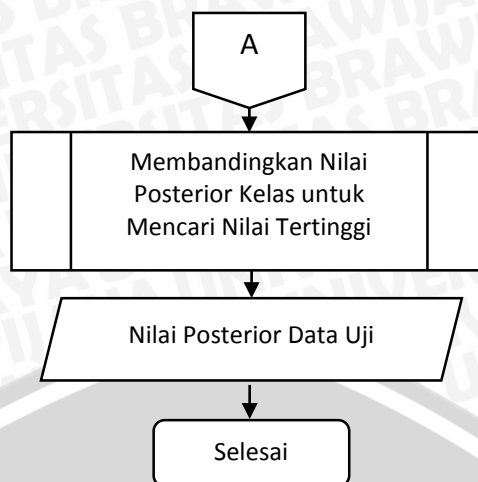
### 3.8.1.5 Menghitung Posterior dari Masing-masing Kemungkinan

Hasil klasifikasi ditentukan dari perbandingan posterior diterima dan tidak dengan menggunakan rumus pada persamaan 2.11. Posterior yang memiliki nilai terbesar akan menjadi hasil klasifikasi. Proses perhitungan posterior dapat dilihat pada Gambar 3.12



Menghitung Posterior dari Masing-Masing Kemungkinan



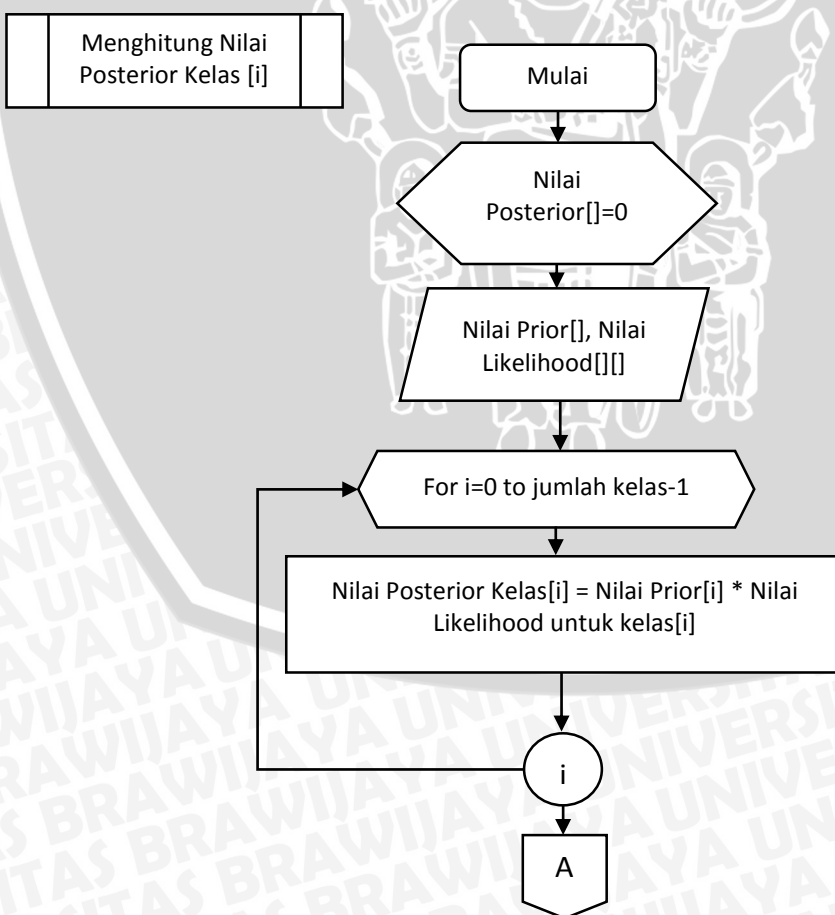


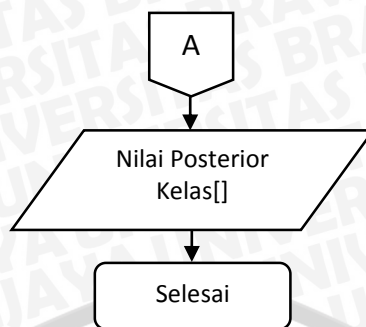
**Gambar 3.12 Diagram Perhitungan Posterior**

Penjelasan alur proses untuk menghitung nilai posterior tiap kelas(kelas diterima dan kelas tidak) adalah sebagai berikut :

a. Menghitung Nilai Posterior Kelas [i]

Proses perhitungan nilai posterior menggunakan persamaan 2.11. Alur proses dari menghitung posterior masing-masing kelas dapat dilihat pada Gambar 3.13





Gambar 3.13 Diagram Perhitungan Nilai Posterior Tiap Kelas

- Posterior diterima :

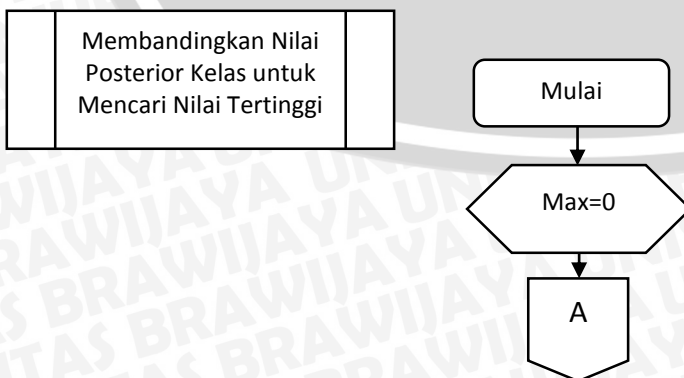
$$\begin{aligned}
 &P(\text{Diterima}) \prod_{j=1}^n P(X_j | \text{Diterima}) \\
 &= P(\text{Diterima}) * P(\text{Matematika}=80 | \text{Diterima}) * P(\text{Fisika}=89 | \text{Diterima}) * \\
 &\quad P(\text{Kimia}=88 | \text{Diterima}) * P(\text{Biologi}=85 | \text{Diterima}) * P(\text{TIK}=86 | \text{Diterima}) * \\
 &\quad P(\text{Bind}=85 | \text{Diterima}) * P(\text{Bing}=87 | \text{Diterima}) \\
 &= 0.5 * 0,018799851 * 0,063107677 * 0,069980505 * 0,068570691 * \\
 &\quad 0,322875813 * 0,175561167 * 0,049778479 \\
 &= \mathbf{0,000000008032}
 \end{aligned}$$

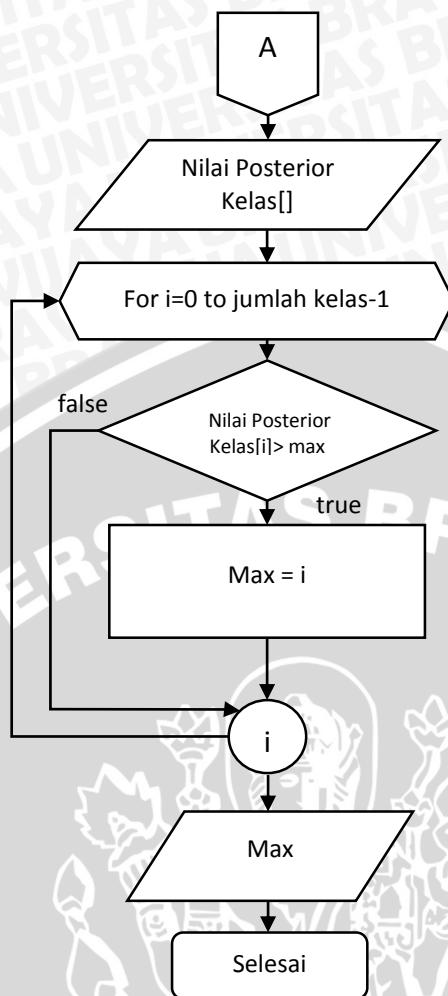
- Posterior tidak :

$$\begin{aligned}
 &P(\text{Tidak}) \prod_{j=1}^n P(X_j | \text{Tidak}) \\
 &= P(\text{Tidak}) * P(\text{Matematika}=80 | \text{Tidak}) * P(\text{Fisika}=89 | \text{Tidak}) * \\
 &\quad P(\text{Kimia}=88 | \text{Tidak}) * P(\text{Biologi}=85 | \text{Tidak}) * P(\text{TIK}=86 | \text{Tidak}) * \\
 &\quad P(\text{Bind}=85 | \text{Tidak}) * P(\text{Bing}=87 | \text{Tidak}) \\
 &= 0.5 * 0,005474315 * 0,001758551 * 0,012649912 * 0,022990857 * \\
 &\quad 0,170177501 * 0,397502938 * 0,025512577 \\
 &= \mathbf{0,000000000441}
 \end{aligned}$$

b. Membandingkan Nilai Posterior Kelas untuk Mencari Nilai Tertinggi

Untuk menghitung nilai posterior suatu data uji, digunakan persamaan 2.12 dengan alur proses yang dapat dilihat pada Gambar 3.14





**Gambar 3.14 Diagram Perbandingan Nilai Posterior tiap Kelas**

$$\begin{aligned}
 P(C|X_1, X_2 \dots X_n) &= \arg \max P(C) \prod_n P(X_1 \dots xX_n | C_i) \\
 &= \arg \max \text{Posterior}(\text{Diterima}) / \text{Posterior}(\text{Tidak}) \\
 &= \arg \max (0,000000008032) / (0,000000000441) \\
 &= 0,000000008032
 \end{aligned}$$

Dicari argument max dari 2 posterior yang telah didefinisikan, yaitu posterior kelas diterima dengan nilai 0,000000008032 dan posterior kelas tidak dengan nilai 0,000000000441. Karena nilai posterior kelas diterima lebih besar daripada nilai posterior kelas tidak, maka hasil klasifikasi dari kasus di atas adalah diterima.

### 3.8.2 Penyelesaian dengan AHP

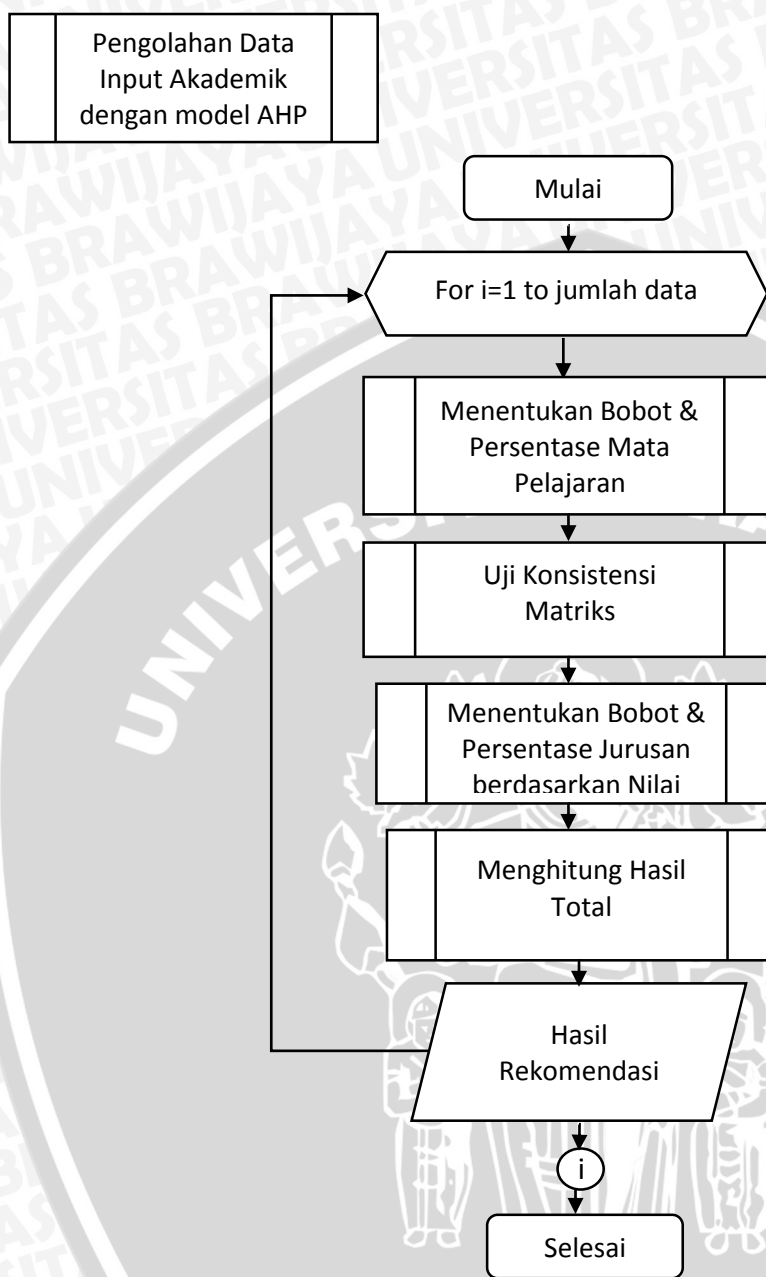
Pada bagian ini akan dibahas perancangan sistem untuk memperoleh rekomendasi jurusan kuliah menggunakan metode AHP. Langkah awal dari bagian ini adalah menentukan bobot dan persentase nilai mata pelajaran dari calon mahasiswa. Pembobotan dilakukan untuk membuat matriks perbandingan berpasangan sehingga persentase mata pelajaran dapat diketahui. Setelah itu dilakukan uji konsistensi matriks agar matriks dapat dinyatakan valid atau tidak.

Proses berikutnya adalah menentukan bobot dan persentase jurusan berdasarkan nilai mahasiswa. Pembobotan jurusan dilihat dari nilai minimum per jurusan dibandingkan dengan nilai calon mahasiswa. Setelah dilakukan pembobotan, dibuatlah matriks perbandingan antar mata pelajaran untuk mengetahui persentase jurusan berdasarkan mata pelajaran tersebut. Dilakukan juga uji konsistensi untuk mengetahui valid atau tidaknya matriks tersebut. Terakhir, sistem akan menentukan persentase total dari tiap jurusan. Jurusan dengan persentase total tertinggi akan menjadi hasil rekomendasi untuk calon mahasiswa. Perancangan model *AHP* sendiri akan menggunakan 3 model untuk pengujian. Ketiga model tersebut memiliki perbedaan parameter di dalam perhitungannya. Detail parameter tiap model dapat dilihat pada Tabel 3.6

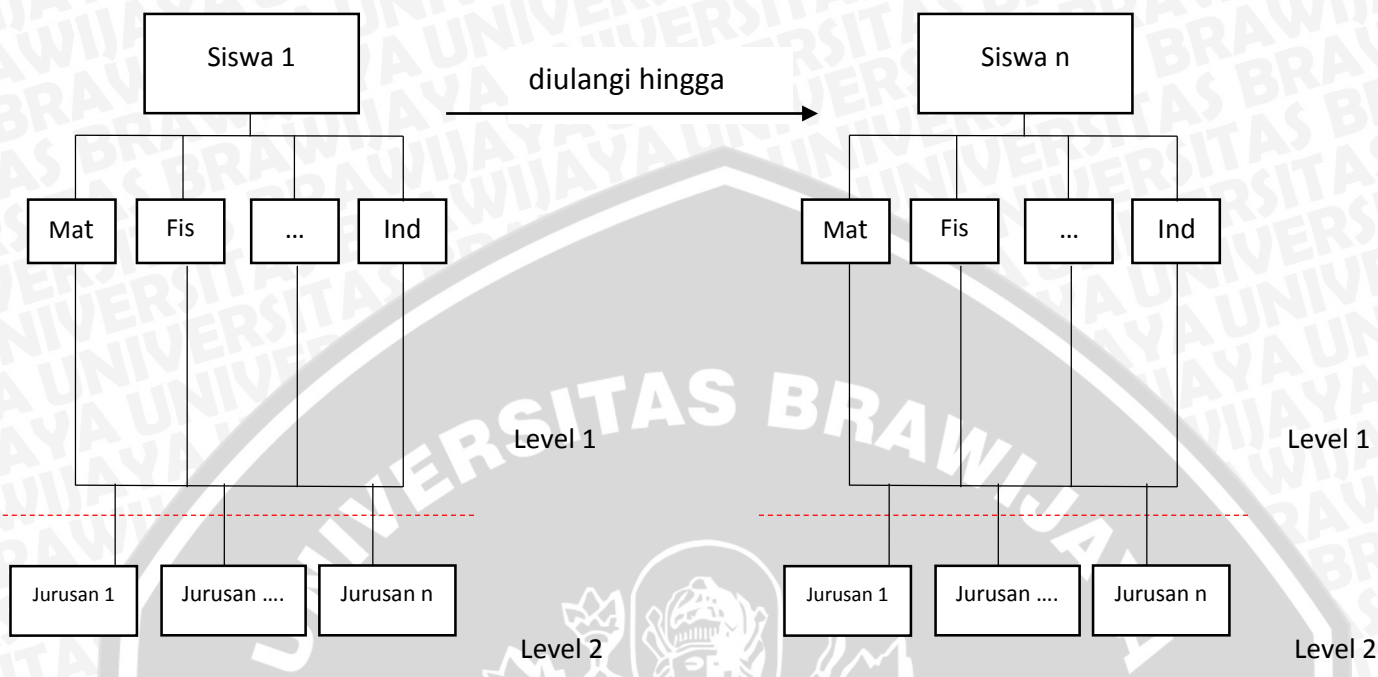
**Tabel 3.6 Parameter Tiap Model AHP**

Model 1	Model 2	Model 3
Agama	Agama	Bahasa Indonesia
Kewarganegaraan	Kewarganegaraan	Bahasa Inggris
Bahasa Indonesia	Bahasa Indonesia	Matematika
Bahasa Inggris	Bahasa Inggris	Fisika
Matematika	Sains	Kimia
Fisika	Sejarah	Biologi
Kimia	Seni Budaya	TIK
Biologi	Penjaskes	
Sejarah	TIK	
Seni Budaya	Bahasa Asing	
Penjaskes		
TIK		
Bahasa Asing		

Model 1 melibatkan semua mata pelajaran, meliputi agama, kewarganegaraan, bahasa Indonesia, bahasa Inggris, matematika, fisika, kimia, biologi, sejarah, seni budaya, pendidikan jasmani, TIK, dan kemampuan berbahasa asing. Untuk model 2 memiliki kemiripan dengan model 1, namun terdapat 1 parameter yaitu mata pelajaran sains yang merupakan nilai rata-rata dari mata pelajaran matematika, fisika, kimia, dan biologi. Untuk model 3 hanya mata pelajaran yang dianggap berada di lingkup IPA secara murni, yaitu mata pelajaran matematika, fisika, kimia, biologi, bahasa Indonesia, bahasa Inggris, dan TIK. Ketiga model ini dibuat untuk membuat variasi di dalam perhitungan dan juga untuk mengetahui model yang memberikan hasil rekomendasi terbaik. Perancangan sistem AHP dapat dilihat di Gambar 3.15 dan model sistem AHP dapat dilihat di Gambar 3.16



Gambar 3.15 Diagram Algoritma AHP



**Gambar 3.16 Model Sistem Algoritma AHP**

### 3.8.2.1 Menentukan Bobot dan Persentase Nilai Mata Pelajaran

Mata Pelajaran yang akan dicari bobotnya adalah matematika, fisika, kimia, biologi, TIK, bahasa Indonesia, dan bahasa Inggris. Pembobotan dilakukan dengan menggunakan 5 kriteria, yaitu rendah dengan bobot 1 (nilai sama dengan 75 atau kurang), rendah menengah dengan bobot 2 (nilai 76 - 80), menengah dengan bobot 3 (nilai 81 - 85), menengah utama dengan bobot 4 (nilai 86-90), dan utama dengan bobot 5 (nilai lebih tinggi dari 90). Aturan pembobotan dapat dilihat pada Tabel 3.7

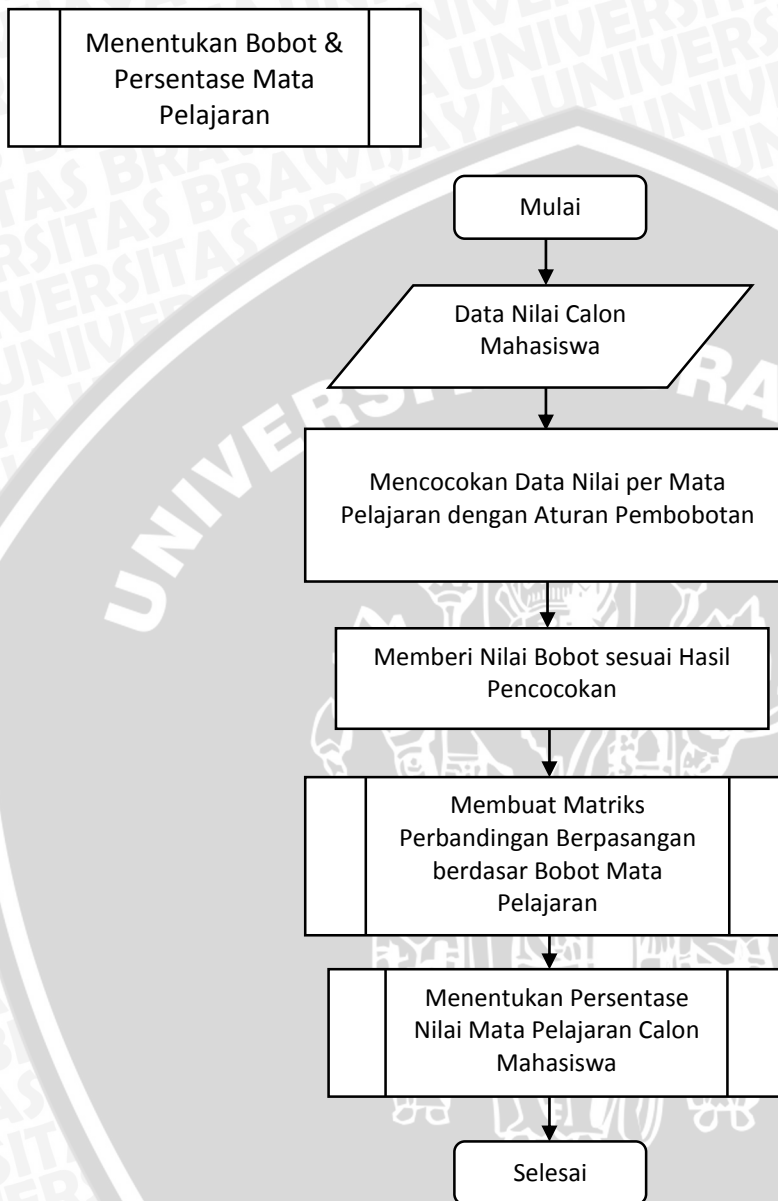
**Tabel 3.7 Aturan Pembobotan**

Nilai	Bobot
Nilai ≤ 75	1
76 ≤ Nilai ≤ 80	2
81 ≤ Nilai ≤ 85	3
86 ≤ Nilai ≤ 90	4
Nilai > 90	5

Dengan data uji yang telah didefinisikan pada Tabel 3.3, maka pembobotan dapat dilakukan. Persentase nilai mata pelajaran ditentukan setelahnya dengan



membuat matriks perbandingan berpasangan. Untuk alur proses dari penentuan bobot dan persentase nilai mata pelajaran dapat dilihat pada Gambar 3.17



**Gambar 3.17 Proses Menentukan Bobot dan Persentase Mata Pelajaran**

Penjelasan dari Gambar 3.17 adalah :

- Mendapatkan Data Nilai Calon Mahasiswa

Data uji yang akan digunakan di dalam perhitungan ini adalah data sample dengan detail yang dijelaskan pada Tabel 3.8

**Tabel 3.8 Data Uji Sampel**

Matematika	Fisika	Kimia	Biologi	TIK	Bind	Bing
80	89	88	78	86	85	87

- b. Mencocokkan Data Nilai per Mata Pelajaran dengan Aturan Pembobotan

Data uji yang telah didefinisikan perlu diberikan bobot. Untuk memberikan bobot, diberikan aturan dengan detail yang dijelaskan pada Tabel 3.9

**Tabel 3.9 Aturan Pembobotan**

Range Nilai	Bobot
Nilai > 75	1
75 <= Nilai <= 80	2
80 < Nilai <= 85	3
85 < Nilai <= 90	4
Nilai > 90	5

- c. Memberi Nilai Bobot sesuai Hasil Pencocokan

Nilai dibobotkan dengan aturan yang sudah didefinisikan di dalam Tabel 3.9. Untuk hasil pembobotan dapat dilihat pada Tabel 3.10 dan penjelasan tentang keterangan bobot dapat dilihat pada Tabel 3.11.

**Tabel 3.10 Pembobotan Data Uji Sampel**

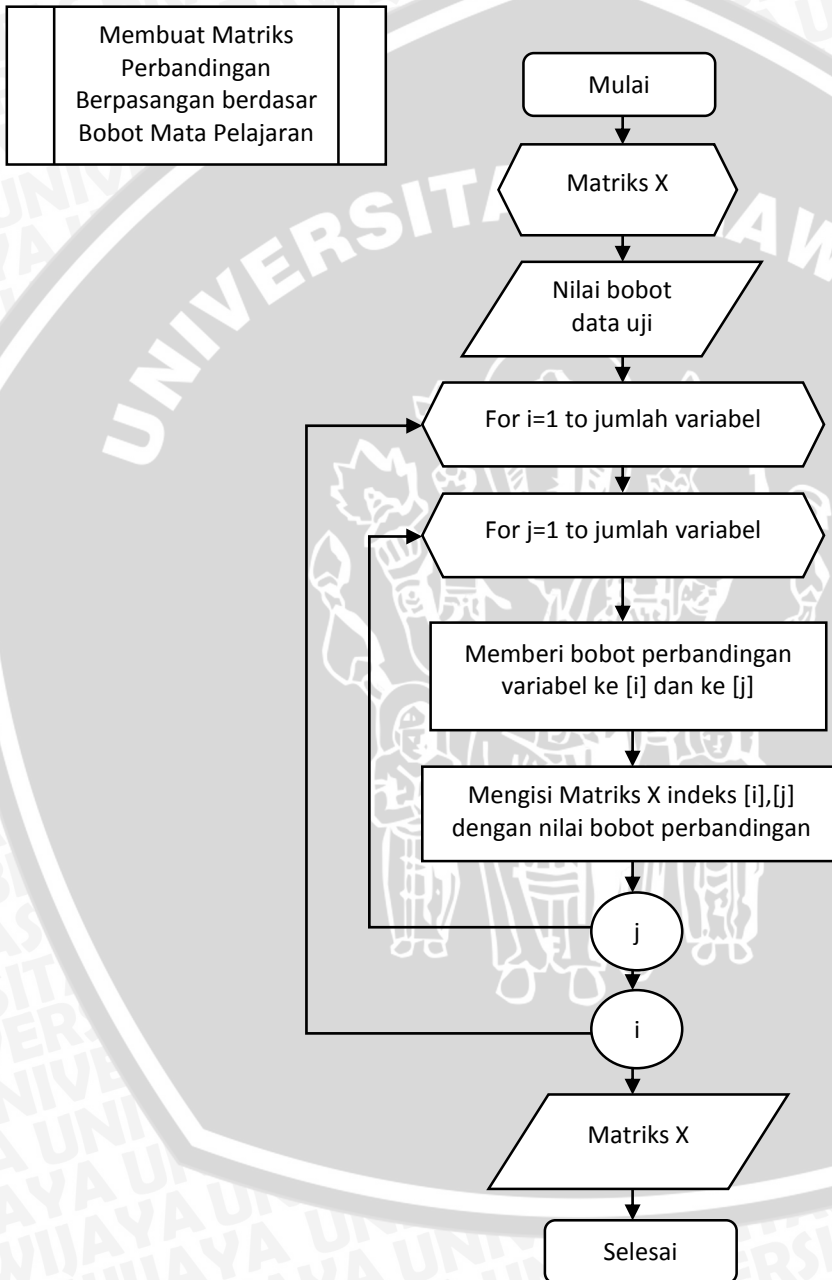
Matematika	Fisika	Kimia	Biologi	TIK	Bind	Bing
80	89	88	78	86	85	87
2	4	4	3	4	3	4

**Tabel 3.11 Keterangan Bobot**

Bobot	Keterangan
1	Rendah
2	Rendah Menengah
3	Menengah
4	Menengah Utama
5	Utama

d. Membuat Matriks Perbandingan Berpasangan berdasar Bobot Mata Pelajaran

Setelah bobot selesai ditentukan, maka proses berikutnya adalah membuat matriks perbandingan berpasangan. Pembuatan matriks perbandingan berpasangan dilakukan berdasarkan bobot masing-masing nilai mata pelajaran. Untuk alur proses dari pembuatan matriks perbandingan berpasangan dapat dilihat pada Gambar 3.18



**Gambar 3.18 Pembuatan Matriks Perbandingan Berpasangan**

Perbandingan antar 2 nilai mata pelajaran bernilai 1 jika memiliki bobot yang sama, bernilai 3 jika memiliki selisih 1 bobot, bernilai 5 jika memiliki selisih 2 bobot,

bernilai 7 jika memiliki selisih 3 bobot, dan bernilai 9 jika memiliki selisih 4 bobot. Jika bobot mata pelajaran yang dibandingkan lebih besar daripada mata pelajaran acuan, maka nilai bobot nya adalah 1/bobot. Aturan di dalam pengisian matriks dapat dilihat pada Tabel 3.12

**Tabel 3.12 Aturan Pembuatan Matriks**

Nilai	Keterangan
1	Sama
3	Berselisih 1
5	Berselisih 2
7	Berselisih 3
9	Berselisih 4

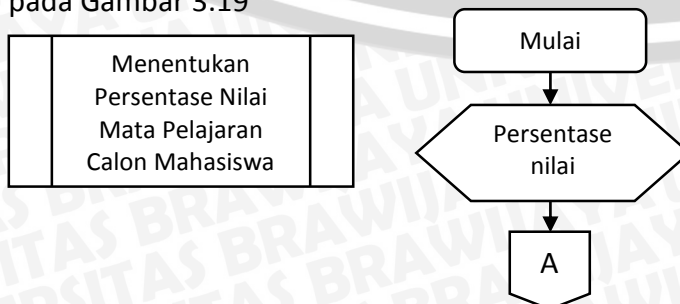
Contoh perhitungan adalah perbandingan mata pelajaran matematika dengan fisika, yang memiliki bobot masing-masing 2 dan 4. Dikarenakan selisih bobot mencapai 2, maka hasil perbandingan berpasangan untuk matematika dibandingkan dengan fisika adalah 1/5. Keterangan nilai matriks dapat dilihat pada Tabel 3.9, dan hasil lengkap matriks perbandingan berpasangan berdasarkan data uji dapat dilihat pada Tabel 3.13.

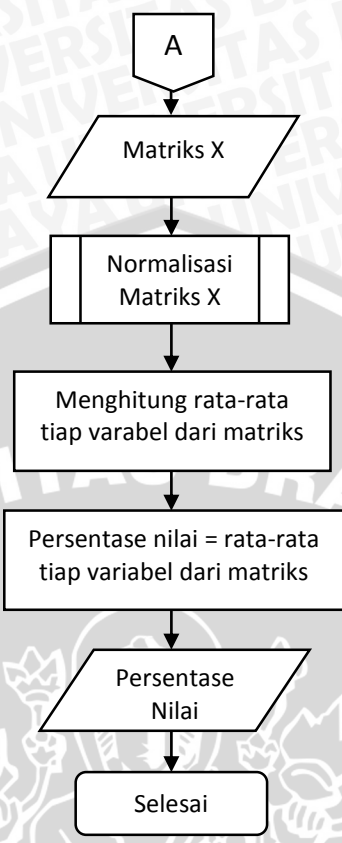
**Tabel 3.13 Matriks Perbandingan Berpasangan**

	Mat	Fis	Kim	Bio	TIK	Bind	Bing
Mat	1	0,2	0,2	0,33	0,2	0,33	0,2
Fis	5	1	1	3	1	3	1
Kim	5	1	1	3	1	5	1
Bio	3	0,33	0,33	1	0,33	1	0,33
TIK	5	1	1	3	1	3	1
Bind	3	0,33	0,2	1	0,33	1	0,33
Bing	5	1	1	3	1	3	1

e. Menentukan Persentase Nilai Mata Pelajaran Calon Mahasiswa

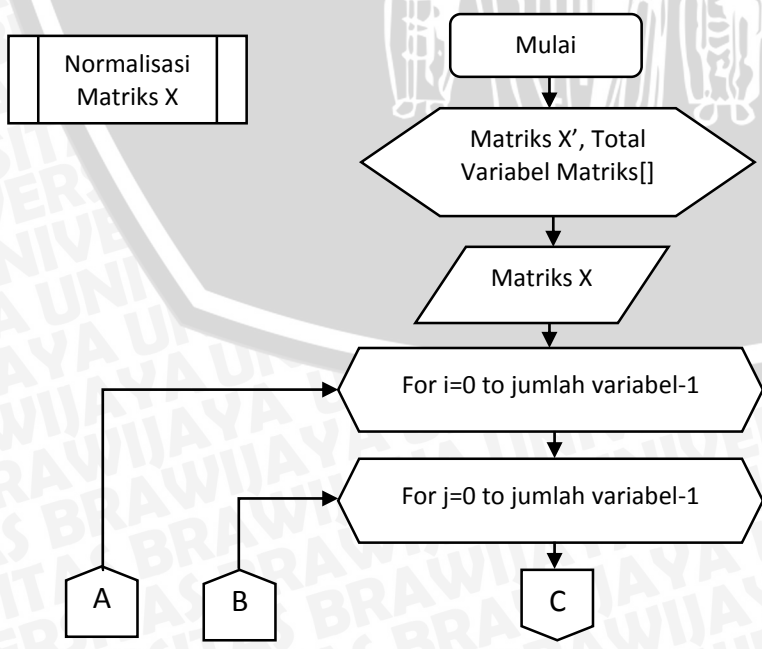
Alur proses penentuan nilai mata pelajaran calon mahasiswa dapat dilihat pada Gambar 3.19

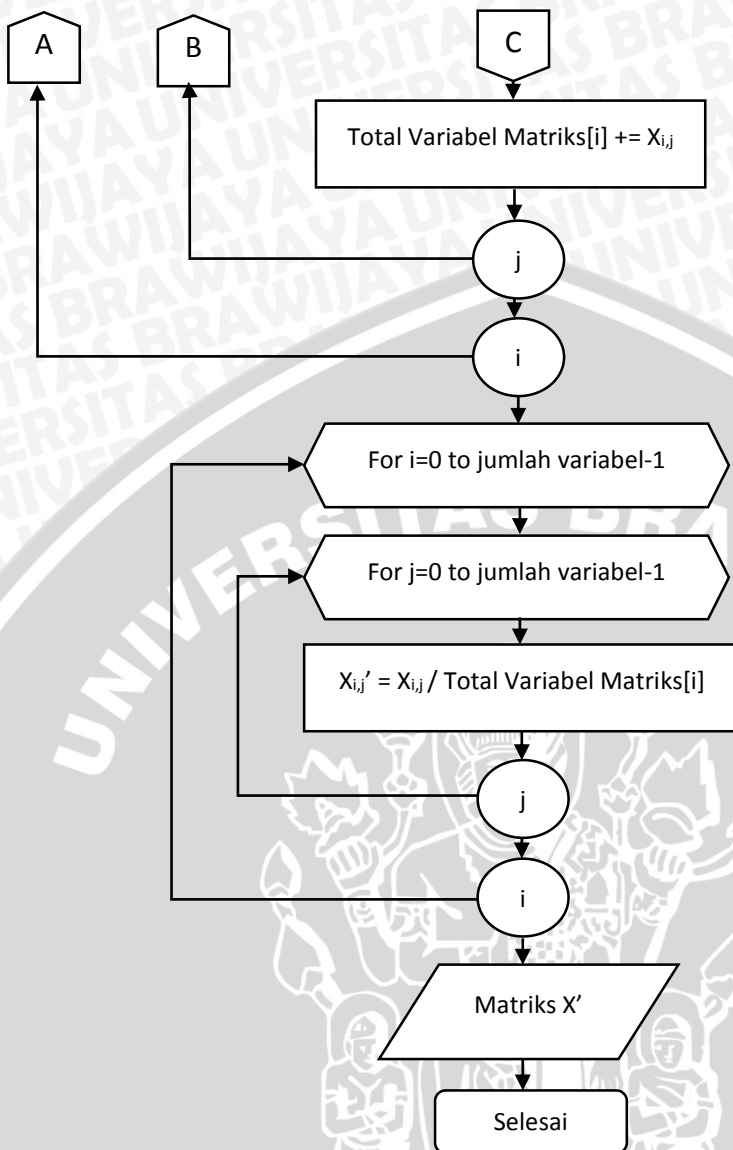




**Gambar 3.19 Penentuan Persentase Nilai Mata Pelajaran Calon Mahasiswa**

Setelah matriks terbentuk, untuk mendapatkan persentase nilai, matriks perlu dinormalisasi terlebih dahulu untuk mendapatkannya. Untuk normalisasi data matriks, digunakan persamaan 2.1 dan untuk mendapatkan bobot kriteria dari suatu parameter digunakan persamaan 2.2. alur proses untuk normalisasi data dapat dilihat pada Gambar 3.20





**Gambar 3.20 Pembuatan Matriks Normalisasi**

Perhitungan normalisasi untuk perbandingan matematika dengan fisika adalah :

$$- X_{2,1}' = \frac{X_{2,1}}{\sum_{j=1}^n X_{2,j}}$$

$$- X_{2,1}' = \frac{0,2}{4,87}$$

$$- X_{2,1}' = 0,04$$

Keterangan

-  $X_{2,1}$  = Nilai Matriks pada indeks (2,1) : 0.2

-  $\sum_{j=1}^n X_{2,j}$  = Total bobot fisika kolom 2: (0,2 + 1 + 1 + 0.33 + 1 + 0.33 + 1)  
= 4,87

-  $X_{2,1}' = \text{Nilai normalisasi indeks (2,1): } 0.2 / 4.87 = 0.04$

Dengan perhitungan tersebut, maka dapat dihitung semua nilai dari matriks normalisasi data. Bobot kriteria pada perhitungan ini dapat dilihat pada kolom rata-rata. Tabel hasil normalisasi data beserta rata-rata dan  $\lambda$  untuk tiap mata pelajaran dapat dilihat pada Tabel 3.14.

**Tabel 3.14 Matriks Normalisasi Data**

	Mat	Fisika	Kimia	Biologi	TIK	Bind	Bing	Total	Rata2	Lamda( $\lambda$ )
Mat	0,04	0,04	0,04	0,02	0,04	0,02	0,04	0,24	0,03	8
Fisika	0,19	0,2	0,2	0,21	0,2	0,19	0,2	1,39	0,2	6,95
Kimia	0,19	0,2	0,2	0,21	0,2	0,31	0,2	1,51	0,22	6,86
Biologi	0,11	0,07	0,07	0,07	0,07	0,06	0,07	0,52	0,07	7,42
TIK	0,19	0,2	0,2	0,21	0,2	0,19	0,2	1,39	0,2	6,95
Bind	0,11	0,07	0,04	0,07	0,07	0,06	0,07	0,49	0,07	7
Bing	0,19	0,2	0,2	0,21	0,2	0,19	0,2	1,39	0,2	6,95

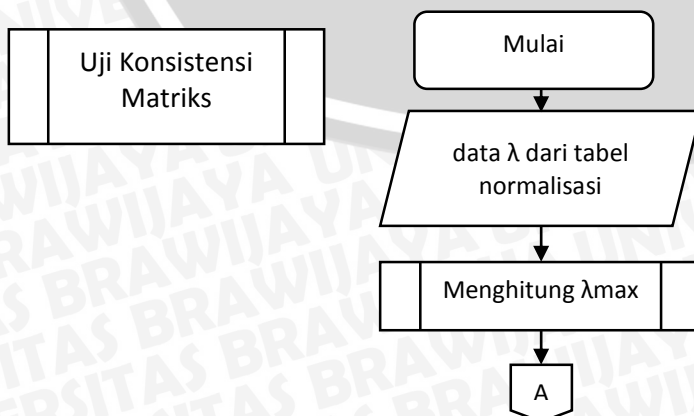
Persentase tiap mata pelajaran dapat dilihat pada Tabel 3.14 pada kolom rata-rata sementara kolom lamda berguna untuk uji konsistensi matriks. Detail persentase mata pelajaran dapat dilihat pada Tabel 3.15

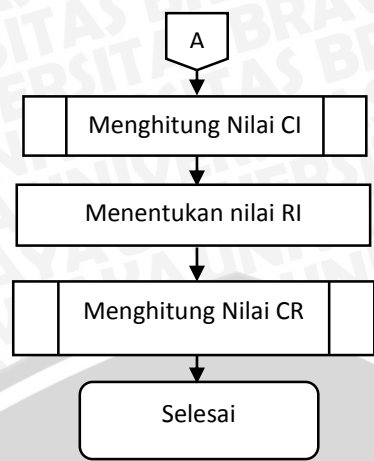
**Tabel 3.15 Detail Persentase Mata Pelajaran**

Mat	Fis	Kim	Bio	Bind	Bing	TIK
0,03	0,2	0,22	0,07	0,2	0,07	0,2

### 3.8.2.2 Uji Konsistensi Matriks

Pengujian konsistensi diperlukan agar didapat hasil rekomendasi yang lebih baik. Matriks bobot yang dihasilkan berdasarkan nilai akademik calon mahasiswa diuji konsistensinya dengan menggunakan persamaan yang sudah didefinisikan pada persamaan 2.3, persamaan 2.4, dan persamaan 2.5. Di dalam perhitungan ini, dibutuhkan nilai rata-rata dan lamda yang dapat dilihat pada Tabel 3.14. Alur proses uji konsistensi matriks dapat dilihat pada Gambar 3.21



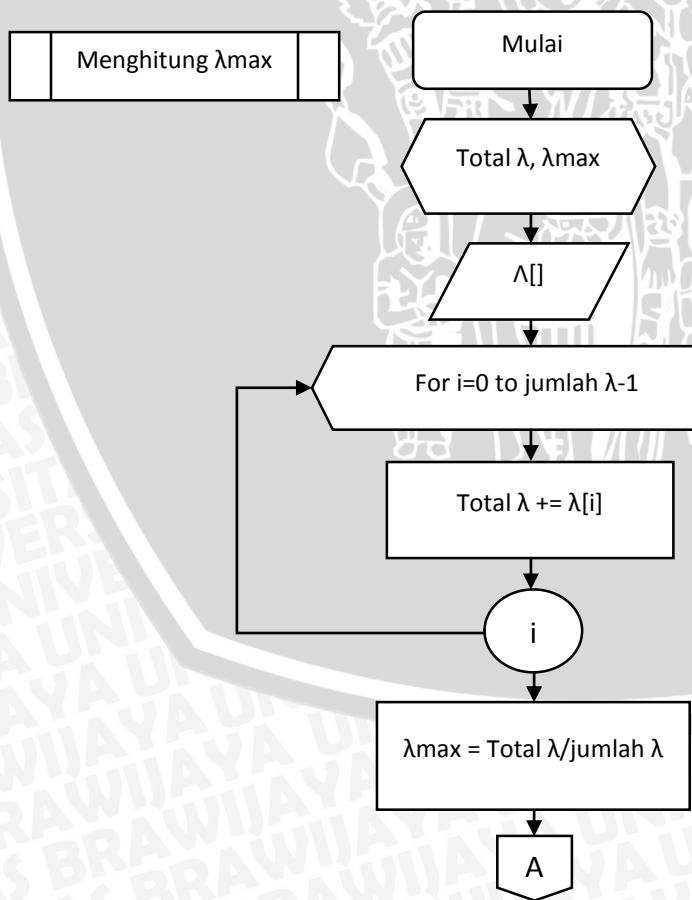


**Gambar 3.21 Proses Uji Konsistensi Matriks**

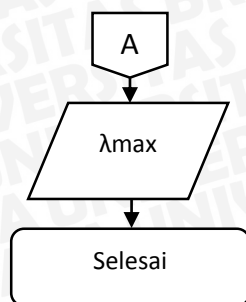
Penjelasan dari Gambar 3.18 adalah :

- a. Mendapatkan data  $\lambda$  dari tabel normalisasi  
Data  $\lambda$ (lamda) dapat dilihat pada Tabel 3.14 pada kolom Lamda.
- b. Menghitung  $\lambda_{max}$

Untuk menghitung  $\lambda_{max}$  menggunakan persamaan 2.3 dengan alur proses dapat dilihat pada Gambar 3.22







**Gambar 3.22 Proses Perhitungan  $\lambda_{max}$**

hasil dari perhitungan  $\lambda_{max}$  dapat dilihat sebagai berikut sebagai berikut :

$$\lambda_{max} = \frac{\sum_{j=1}^n \lambda_j}{n}$$

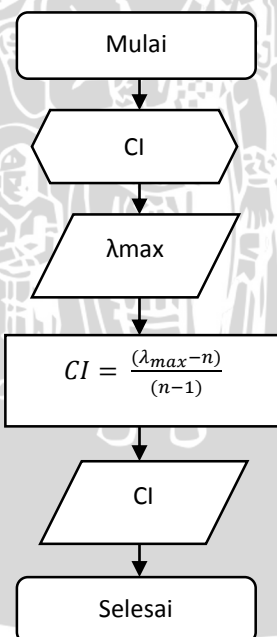
$$\lambda_{max} = (8 + 6,95 + 6,86 + 7,42 + 6,95 + 7 + 6,95) / 7$$

$$= 7,16$$

c. Menghitung Nilai CI

Untuk menghitung Consistency Index (CI) digunakan persamaan 2.4 dengan alur proses yang dapat dilihat pada Gambar 3.23

Menghitung Nilai CI



**Gambar 3.23 Proses Perhitungan CI**

dengan hasil sebagai berikut :

$$CI = \frac{(\lambda_{max} - n)}{(n - 1)}$$



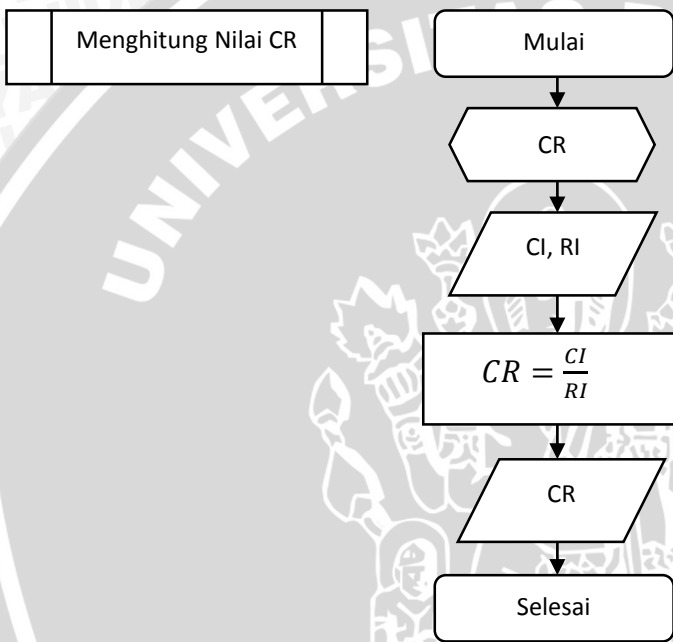
$$CI = (7,16 - 7) / (7-1) = 0,027$$

d. Menentukan nilai RI

Untuk menghitung Ratio Index (RI) digunakan Tabel 2.3 sebagai acuan. Dengan jumlah parameter data adalah 7, sehingga nilai RI = 1.

e. Menghitung Nilai CR

Untuk menghitung Consistency Ratio (CR) digunakan persamaan 2.5 dengan alur proses yang dapat dilihat pada Gambar 3.24



**Gambar 3.24 Proses Perhitungan CR**

dengan hasil sebagai berikut :

$$CR = \frac{CI}{RI}$$

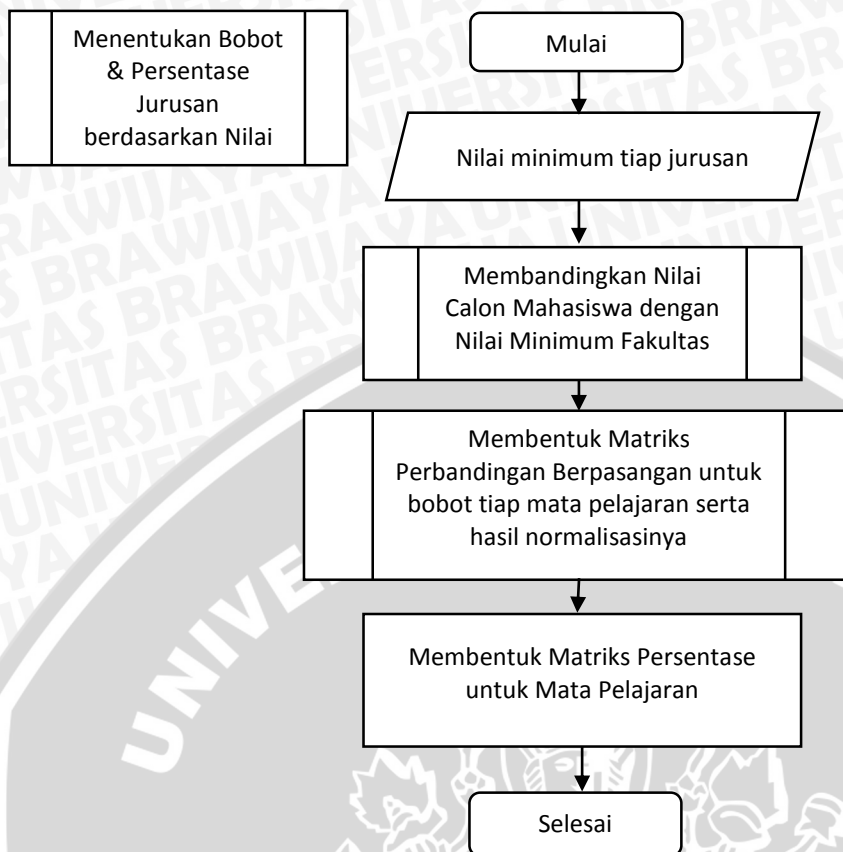
$$CR = 0,027/1 = 0,027$$

Karena nilai CR lebih kecil dari 0.10 , maka matriks dinyatakan konsisten.

### 3.8.2.3 Menentukan Bobot dan Persentase Jurusan berdasarkan Nilai

Penentuan bobot dan persentase jurusan berdasarkan nilai memiliki alur proses yang dapat dilihat pada Gambar 3.25





**Gambar 3.25 Proses Penentuan Bobot dan Persentase Jurusan**

a. Mendapatkan nilai minimum tiap jurusan

Penentuan bobot jurusan dilakukan dengan membandingkan nilai dari calon mahasiswa dengan nilai minimum yang terdapat di tiap jurusan di dalam fakultas tersebut. Di dalam perbandingan ini, akan digunakan 3 jurusan *sample*, yaitu jurusan kedokteran gigi, jurusan gizi, dan jurusan kedokteran. Nilai terendah tiap jurusan dapat dilihat pada Tabel 3.16

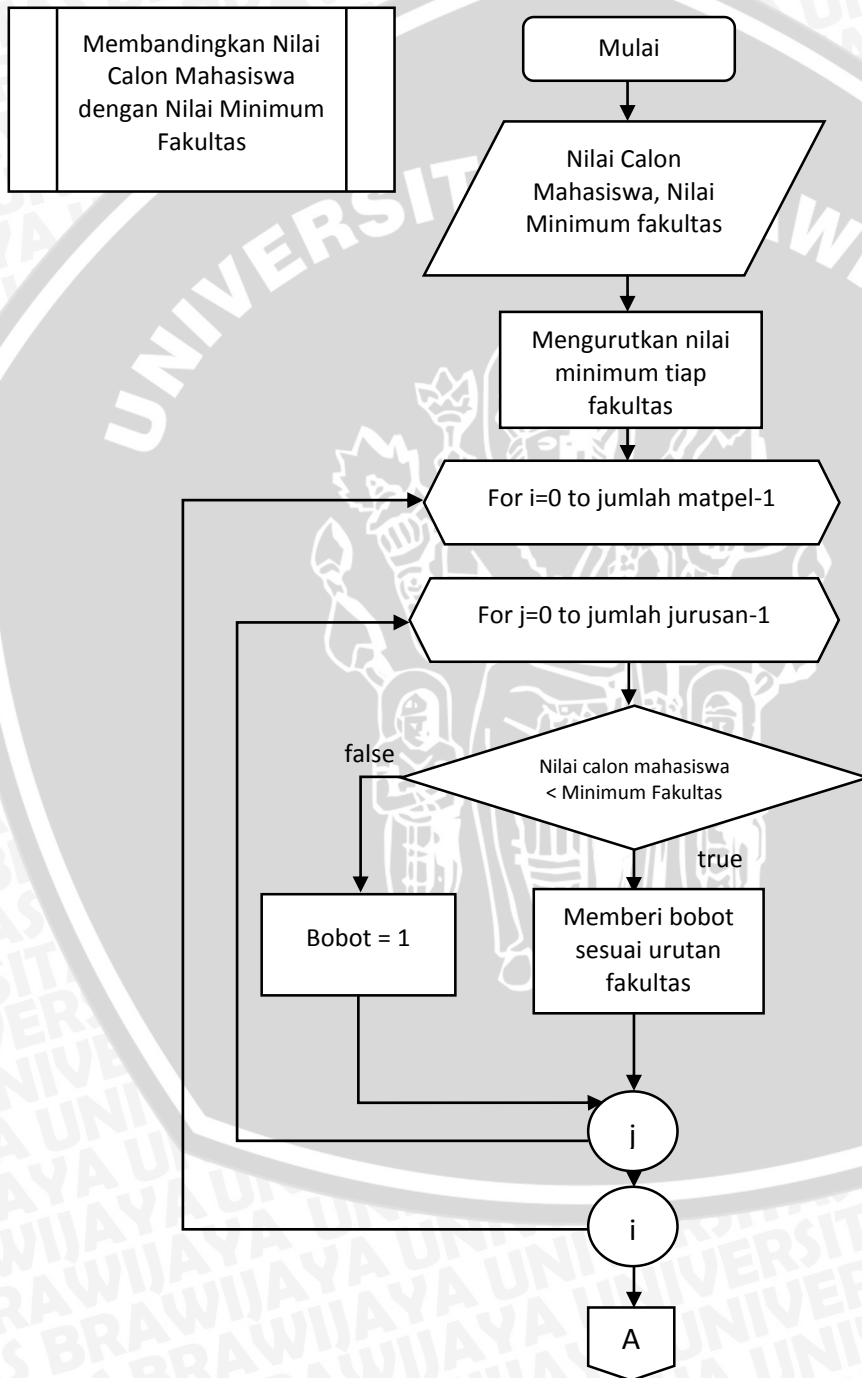
**Tabel 3.16 Matriks Nilai Minimum Jurusan**

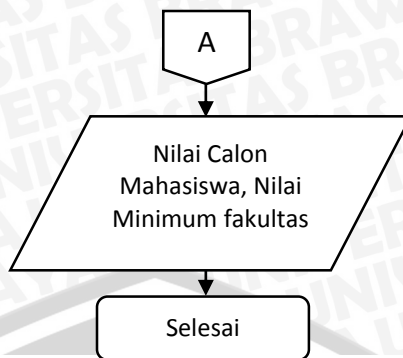
JURUSAN	Mat	Fisika	Kimia	Biologi	TIK	Bind	Bing
Kedokteran Gigi	82	79,66	80,33	85,33	84,66	85,66	81
Gizi	88	82	83	89	86	86	83
Kedokteran	90	85	82	87	83	84	82

b. Membandingkan Nilai Calon Mahasiswa dengan Nilai Minimum Fakultas

Metode perbandingan nilai calon mahasiswa dengan nilai minimum fakultas adalah jurusan yang memiliki nilai minimum yang paling mendekati dengan nilai calon mahasiswa akan memiliki bobot tertinggi. Sebagai contoh, jika nilai calon

mahasiswa adalah 84, sementara nilai minimum jurusan gizi adalah 83, dan nilai minimum jurusan kedokteran adalah 81, maka bobot jurusan gizi lebih tinggi daripada jurusan kedokteran karena nilai calon mahasiswa lebih dekat dengan nilai minimum dari jurusan gizi. Aturan ini juga memiliki syarat tambahan yaitu bobot jurusan akan lebih tinggi selama nilai calon mahasiswa masih lebih tinggi daripada nilai minimum jurusan, jika nilai calon mahasiswa lebih rendah dari nilai minimum jurusan, maka bobot jurusan tersebut akan diberi bobot 1. Alur proses perbandingan secara diagram dapat dilihat pada Gambar 3.26





**Gambar 3.26** Proses Perbandingan Nilai Calon Mahasiswa dengan Nilai Minimum Fakultas

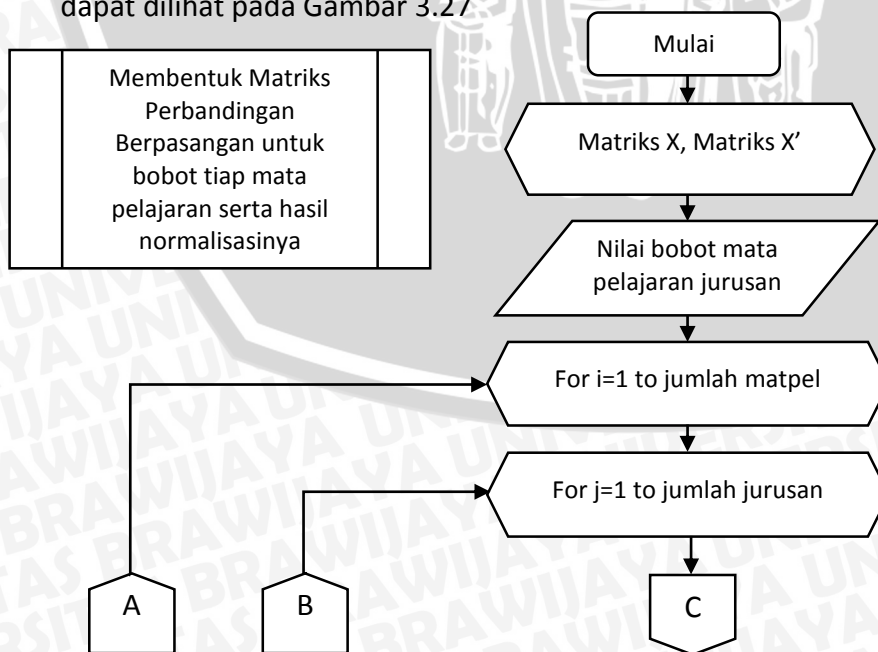
Hasil pembobotan jurusan berdasarkan perbandingan nilai calon mahasiswa dengan nilai minimum tiap jurusan dapat dilihat pada Tabel 3.17:

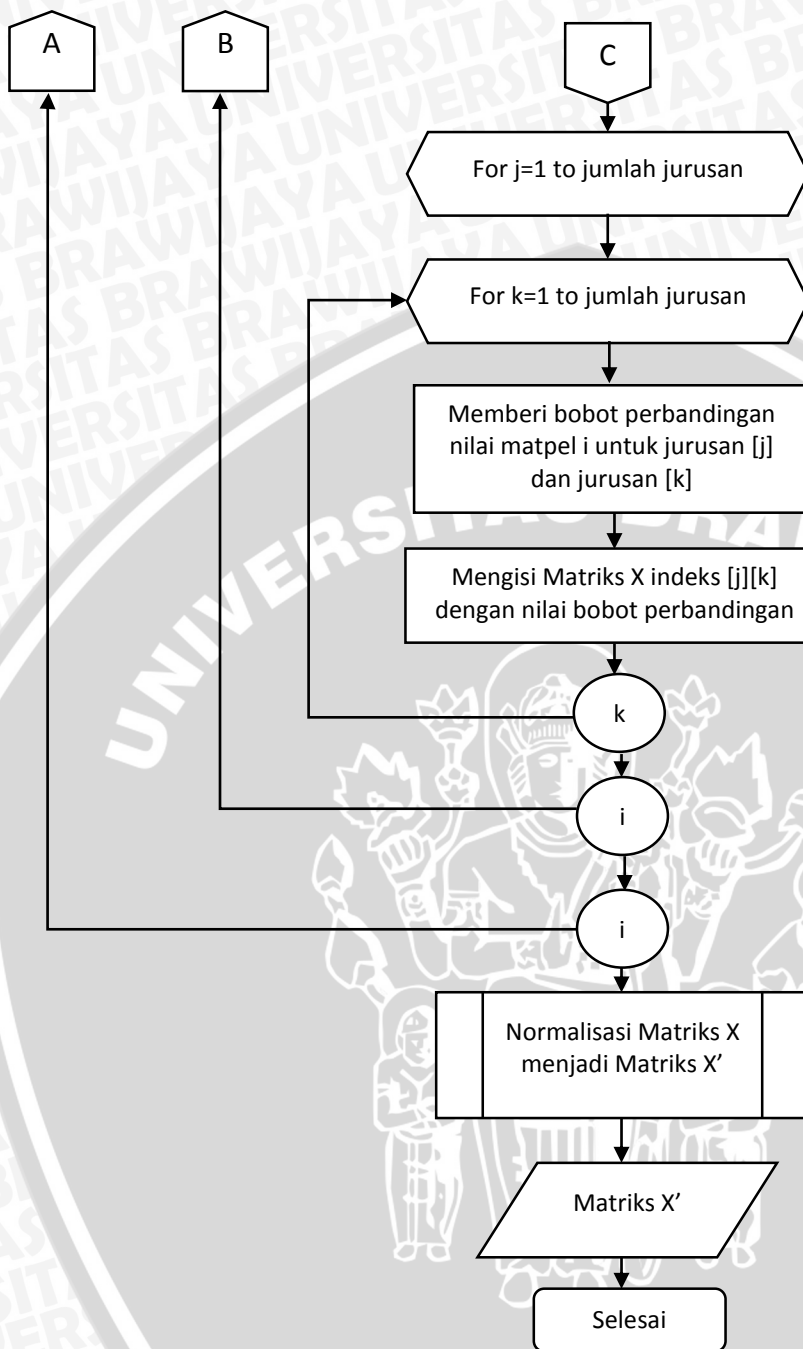
**Tabel 3.17** Matriks Pembobotan Jurusan

JURUSAN	Mat	Fisika	Kimia	Biologi	TIK	Bind	Bing
Kedokteran Gigi	1	1	1	1	2	1	1
Gizi	1	2	3	1	3	1	3
Kedokteran	1	3	2	1	1	2	2

- c. Membentuk Matriks Perbandingan Berpasangan untuk bobot tiap mata pelajaran serta hasil normalisasinya

Menentukan persentase jurusan dilakukan dengan melakukan perhitungan persentase tiap nilai mata pelajaran berdasarkan bobot jurusan. Alur proses untuk membentuk matriks perbandingan berpasangan untuk bobot tiap mata pelajaran dapat dilihat pada Gambar 3.27





**Gambar 3.27 Proses Pembuatan Matriks Perbandingan Berpasangan dan Matriks Normalisasi**

Dan untuk alur normalisasi matriks dapat dilihat pada Gambar 3.12. Persentase tiap nilai mata pelajaran dihitung dengan membuat matriks perbandingan berpasangan yang kemudian dinormalisasikan, dengan hasil dapat dilihat pada Tabel 3.18 hingga Tabel 3.31.

**Tabel 3.18 Matriks Perbandingan Matematika**

Matematika	Kedokteran G	Gizi	Kedokteran
Kedokteran G	1	1	1
Gizi	1	1	1
Kedokteran	1	1	1
Total	3	3	3

**Tabel 3.19 Normalisasi Matriks Perbandingan Matematika**

Matematika	Kedokteran G	Gizi	Kedokteran	Rata2
Kedokteran G	0,33	0,33	0,33	0,33
Gizi	0,33	0,33	0,33	0,33
Kedokteran	0,33	0,33	0,33	0,33

**Tabel 3.20 Matriks Perbandingan Fisika**

Fisika	Kedokteran G	Gizi	Kedokteran
Kedokteran G	1	0,33	0,2
Gizi	3	1	0,33
Kedokteran	5	3	1
Total	9	4,33	1,53

**Tabel 3.21 Normalisasi Matriks Perbandingan Fisika**

Fisika	Kedokteran G	Gizi	Kedokteran	Rata2
Kedokteran G	0,11	0,08	0,13	0,11
Gizi	0,33	0,23	0,22	0,26
Kedokteran	0,56	0,69	0,65	0,63

**Tabel 3.22 Matriks Perbandingan Kimia**

Kimia	Kedokteran G	Gizi	Kedokteran
Kedokteran G	1	0,2	0,33
Gizi	5	1	3
Kedokteran	3	0,33	1
Total	9	1,53	4,33

**Tabel 3.23 Normalisasi Matriks Perbandingan Kimia**

Kimia	Kedokteran G	Gizi	Kedokteran	Rata2
Kedokteran G	0,11	0,13	0,08	0,11
Gizi	0,56	0,65	0,69	0,63
Kedokteran	0,33	0,22	0,23	0,26

**Tabel 3.24 Matriks Perbandingan Biologi**

Biologi	Kedokteran G	Gizi	Kedokteran
Kedokteran G	1	1	1
Gizi	1	1	1
Kedokteran	1	1	1
Total	3	3	3

**Tabel 3.25 Normalisasi Matriks Perbandingan Biologi**

Biologi	Kedokteran G	Gizi	Kedokteran	Rata2
Kedokteran G	0,33	0,33	0,33	0,33
Gizi	0,33	0,33	0,33	0,33
Kedokteran	0,33	0,33	0,33	0,33

**Tabel 3.26 Matriks Perbandingan TIK**

TIK	Kedokteran G	Gizi	Kedokteran
Kedokteran G	1	0,33	3
Gizi	3	1	5
Kedokteran	0,33	0,2	1
Total	4,33	1,53	9

**Tabel 3.27 Normalisasi Matriks Perbandingan TIK**

TIK	Kedokteran G	Gizi	Kedokteran	Rata2
Kedokteran G	0,23	0,22	0,33	0,26
Gizi	0,69	0,65	0,56	0,63
Kedokteran	0,08	0,13	0,11	0,11

**Tabel 3.28 Matriks Perbandingan Bahasa Indonesia**

Bind	Kedokteran G	Gizi	Kedokteran
Kedokteran G	1	1	0,33
Gizi	1	1	0,33
Kedokteran	3	3	1
Total	5	5	1,66

**Tabel 3.29 Normalisasi Matriks Perbandingan Bahasa Indonesia**

Bind	Kedokteran G	Gizi	Kedokteran	Rata2
Kedokteran G	0,2	0,2	0,2	0,2
Gizi	0,2	0,2	0,2	0,2



Kedokteran	0,6	0,6	0,6	0,6
------------	-----	-----	-----	-----

**Tabel 3.30 Matriks Perbandingan Bahasa Inggris**

Bing	Kedokteran G	Gizi	Kedokteran
Kedokteran G	1	0,2	0,33
Gizi	5	1	3
Kedokteran	3	0,33	1
Total	9	1,53	4,33

**Tabel 3.31 Normalisasi Matriks Perbandingan Bahasa Inggris**

Bing	Kedokteran G	Gizi	Kedokteran	Rata2
Kedokteran G	0,11	0,13	0,08	0,11
Gizi	0,56	0,65	0,69	0,63
Kedokteran	0,33	0,22	0,23	0,26

d. Membentuk Membentuk Matriks Persentase untuk Mata Pelajaran

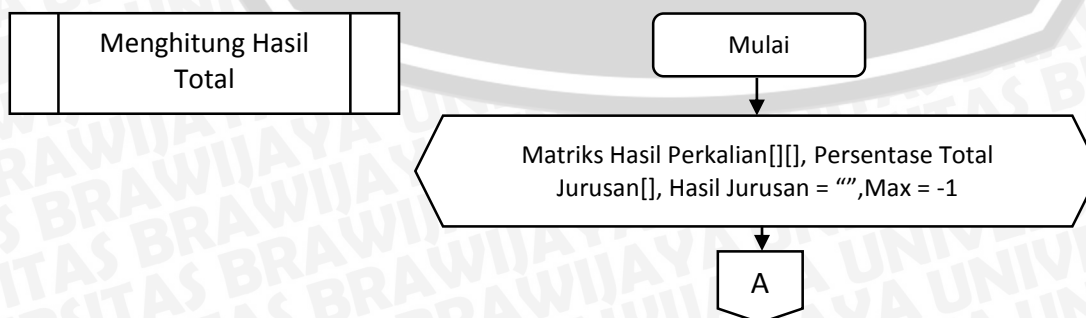
Persentase mata pelajaran didapat dari nilai rata-rata jurusan yang didapat dari tabel normalisasi, yang kemudian disusun menjadi matriks persentase mata pelajaran. Hasil matriks persentase mata pelajaran dapat dilihat pada Tabel 3.32

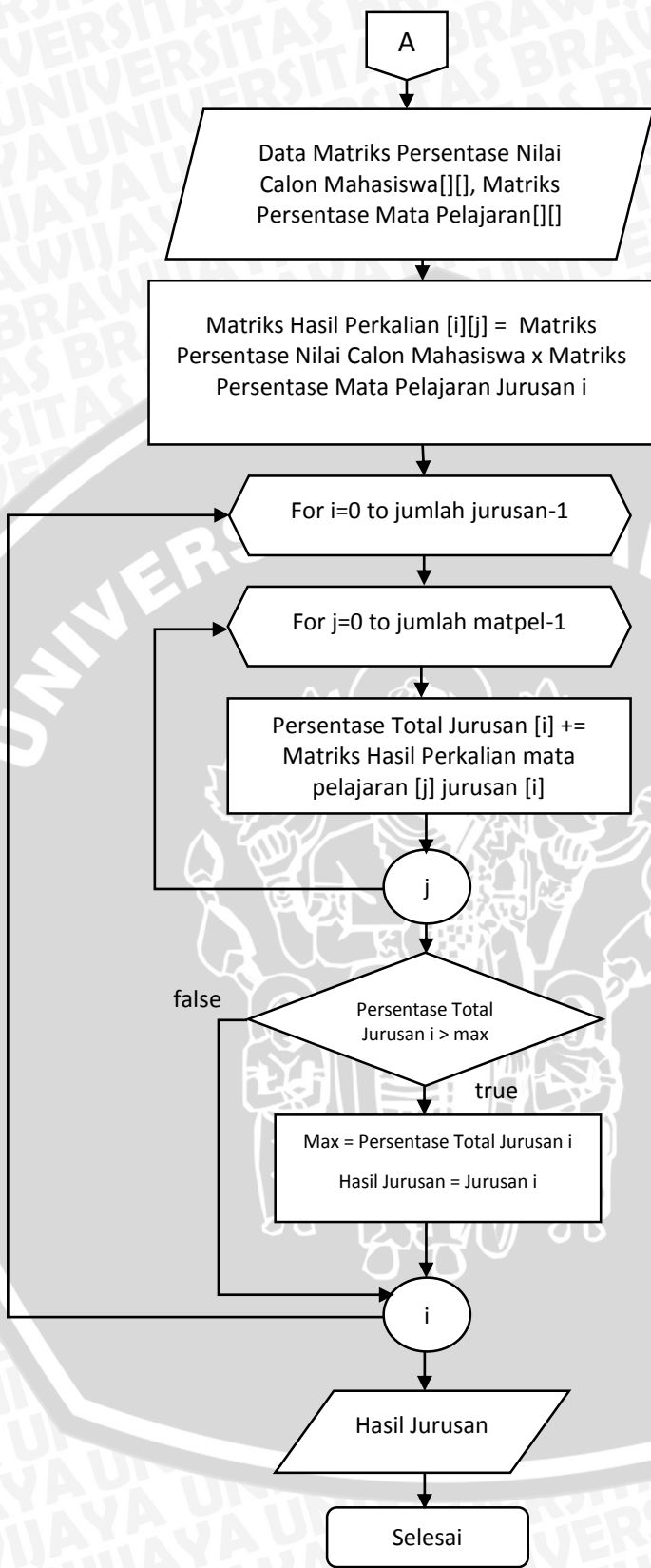
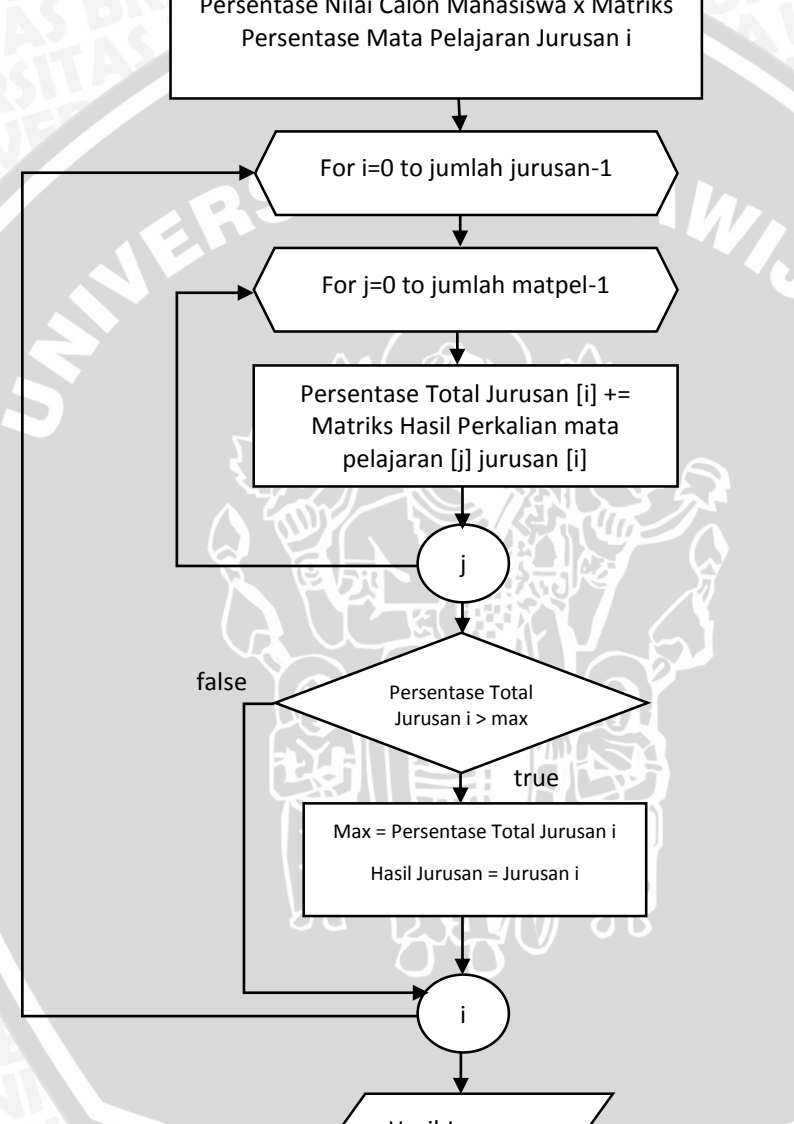
**Tabel 3.32 Matriks Persentase Mata Pelajaran**

JURUSAN	Mat	Fisika	Kimia	Biologi	TIK	Bind	Bing
Kedokteran Gigi	0,33	0,11	0,11	0,33	0,26	0,2	0,11
Gizi	0,33	0,26	0,63	0,33	0,63	0,2	0,63
Kedokteran	0,33	0,63	0,26	0,33	0,11	0,6	0,26

**3.8.2.4 Menghitung Hasil Total**

Menghitung persentase total tiap jurusan dilakukan dengan mengalikan nilai persentase tiap mata pelajaran calon mahasiswa yang terdapat pada Tabel 3.14 kolom rata-rata dengan matriks persentase mata pelajaran per jurusan pada Tabel 3.32. Untuk keterangan diagram dapat dilihat pada Gambar 3.28





Gambar 3.28 Proses Menghitung Hasil Total



Penjelasan dari alur proses di atas adalah :

- a. Input Data Matriks Persentase Nilai Calon Mahasiswa dan Matriks Persentase Mata Pelajaran

Data matriks persentase nilai calon mahasiswa didapatkan dari rata-rata variabel tabel hasil normalisasi pada Tabel 3.14 dan matriks mata pelajaran didapatkan pada Tabel 3.32

- b. Mengalikan Matriks Persentase Nilai Calon Mahasiswa dengan Matriks Persentase Mata Pelajaran

Contoh perhitungan diambil dari mata pelajaran matematika untuk jurusan kedokteran gigi, dengan rata-rata bobot mata pelajaran matematika = 0,03 dan persentase mata pelajaran untuk matematika jurusan kedokteran gigi = 0,33, sehingga persentase total mata pelajaran matematika untuk jurusan kedokteran gigi adalah :  $0,03 \times 0,33 = 0,0099$ . Kemudian untuk mendapatkan persentase total suatu jurusan, tambahkan hasil persentase mata pelajaran yang telah dihitung untuk tiap jurusan. Untuk jurusan kedokteran gigi memiliki persentase total = 0,1672. Hasil persentase total untuk seluruh mata pelajaran tiap jurusan dapat dilihat pada Tabel 3.33.

**Tabel 3.33 Matriks Persentase Total Mata Pelajaran**

JURUSAN	Mat	Fisika	Kimia	Biologi	TIK	Bind	Bing	Total
Kedokteran Gigi	0,0099	0,022	0,0242	0,0231	0,052	0,014	0,022	0,1672
Gizi	0,0099	0,052	0,1386	0,0231	0,126	0,014	0,126	0,4896
Kedokteran	0,0099	0,126	0,0572	0,0231	0,022	0,042	0,052	0,3322

Kemudian setelah diketahui persentase total dari tiap jurusan, dilakukan perbandingan untuk mengetahui jurusan dengan persentase total terbesar. Dari Tabel di atas dapat dilihat bahwa jurusan dengan persentase total terbesar adalah jurusan Gizi, sehingga hasil rekomendasi jurusan untuk calon mahasiswa tersebut adalah Gizi.

## BAB 4 PERANCANGAN

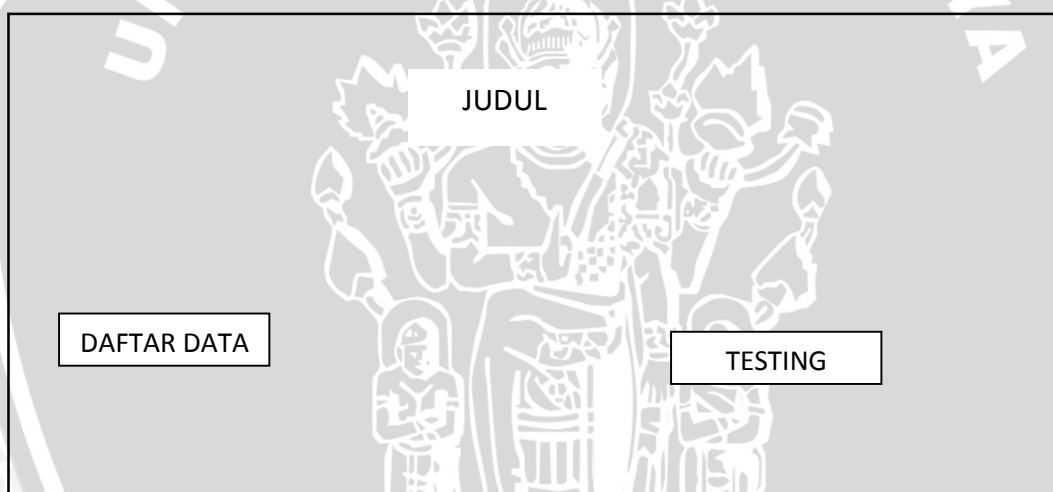
Pada bab ini akan membahas mengenai perancangan sistem, yang meliputi perancangan *user interface*, perancangan alur sistem, dan perancangan pengujian sistem

### 4.1 Perancangan *User Interface*

Pada bagian ini membahas perancangan *user interface* sistem sebanyak 7 halaman, meliputi halaman utama, halaman daftar data, Halaman *setting* data uji, Halaman *testing* Naïve Bayes, Halaman manualisasi *Naïve Bayes*, Halaman *testing* *AHP*, dan halaman manualisasi *AHP*.

#### 4.1.1 Perancangan Halaman Utama

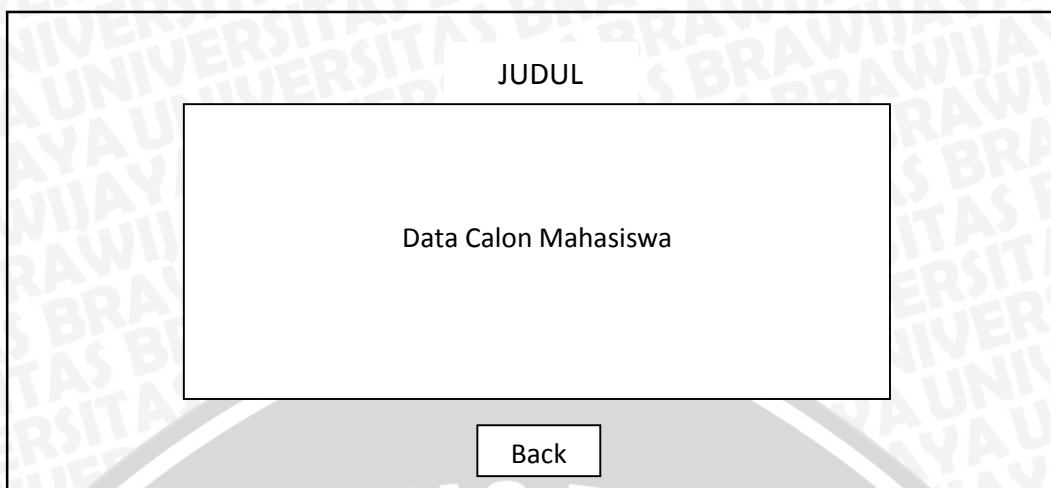
Perancangan halaman utama aplikasi menampilkan judul dari aplikasi, button daftar data untuk menampilkan daftar data yang digunakan di dalam perhitungan sistem, button *testing* untuk menampilkan halaman *testing* Naïve Bayes. Perancangan halaman utama dapat dilihat pada Gambar 4.1.



Gambar 4.1 Perancangan Halaman Utama

#### 4.1.2 Perancangan Daftar Data

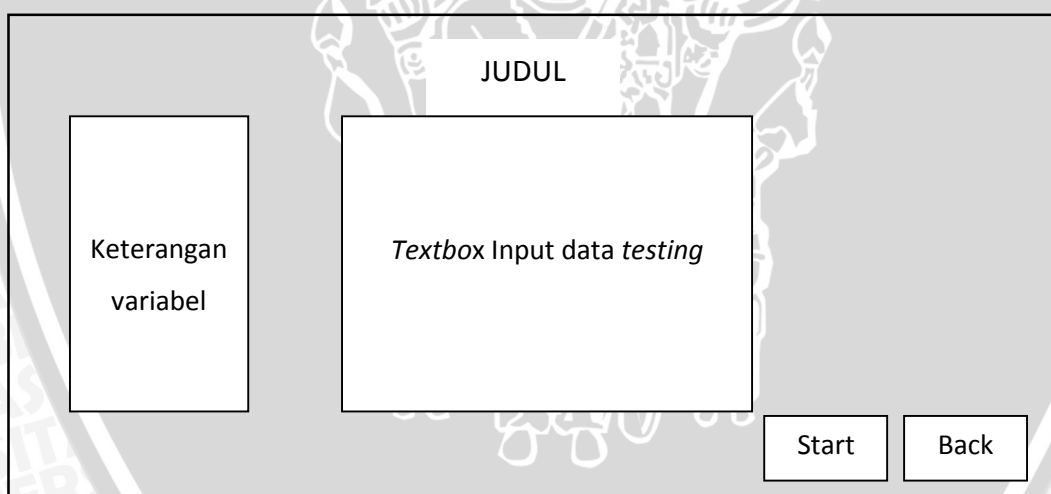
Perancangan halaman daftar data menampilkan judul halaman, beserta data yang akan digunakan dalam proses perhitungan sistem, dan tombol *back* untuk kembali ke halaman utama. Data yang ditampilkan adalah data calon mahasiswa dengan memiliki atribut nilai dari masing-masing mata pelajaran beserta status diterima atau tidaknya calon mahasiswa tersebut di fakultas tertentu. Perancangan halaman daftar data dapat dilihat pada Gambar 4.2.



Gambar 4.2 Perancangan Halaman Daftar Data

#### 4.1.3 Perancangan *Setting* Data Uji

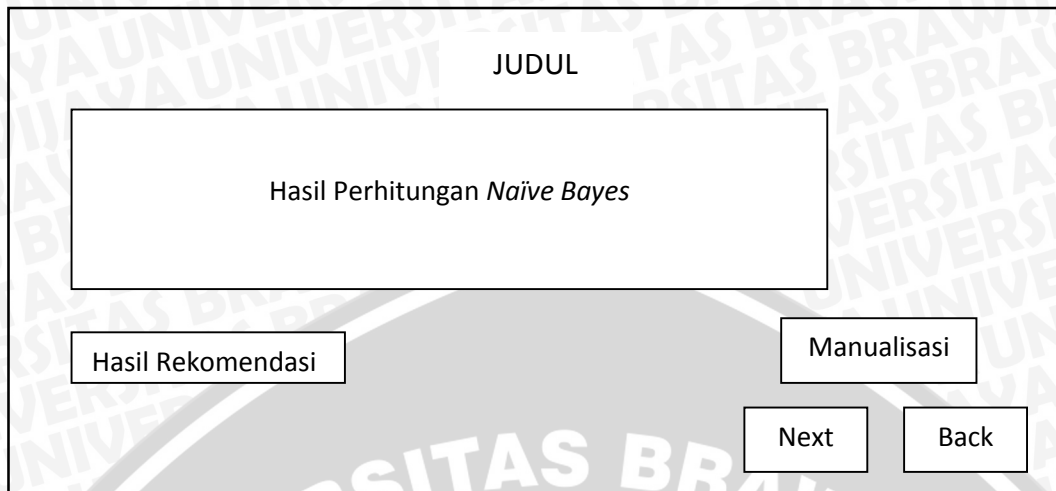
Perancangan *setting* data uji menampilkan judul halaman, *textbox* input data *testing* untuk menginputkan data uji yang akan digunakan di dalam proses perhitungan sistem, *button* Start berfungsi untuk menuju halaman *testing Naive Bayes*, *button* back untuk kembali ke halaman menu utama. Perancangan halaman *setting* data uji dapat dilihat pada Gambar 4.3.



Gambar 4.3 Perancangan Halaman Setting Data Uji

#### 4.1.4 Perancangan *Testing Naive Bayes*

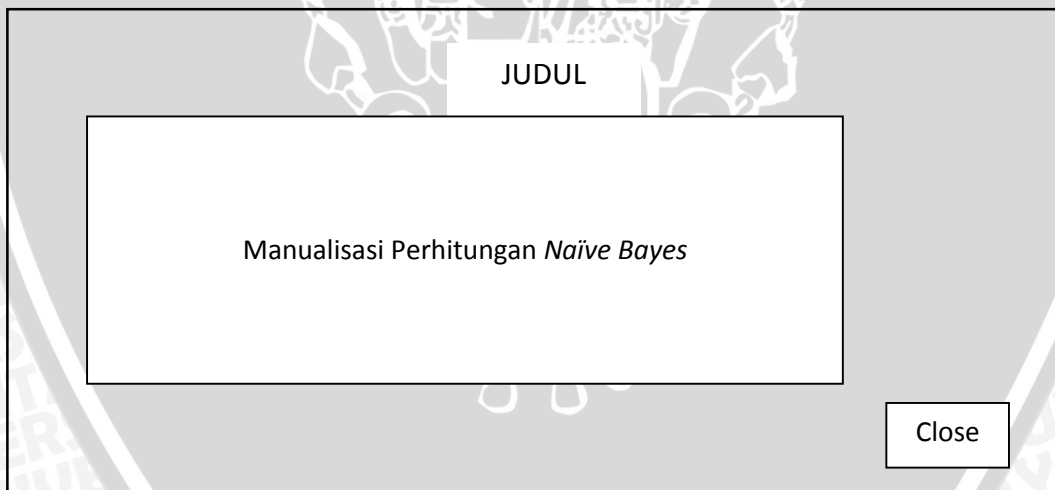
Perancangan *testing naive bayes* menampilkan judul halaman, hasil perhitungan *naive bayes* menampilkan peluang calon mahasiswa diterima di masing-masing fakultas, hasil rekomendasi fakultas menampilkan hasil rekomendasi fakultas untuk calon mahasiswa berdasarkan fakultas dengan tingkat diterima yang paling tinggi, *button* Next untuk menampilkan halaman *testing AHP*, *button* manualisasi untuk menampilkan halaman manualisasi *Naive Bayes*, *button* back untuk kembali ke halaman *setting* data uji. Perancangan *testing naive bayes* dapat dilihat pada Gambar 4.4.



Gambar 4.4 Perancangan Halaman Testing Naive Bayes

#### 4.1.5 Perancangan Manualisasi Naive Bayes

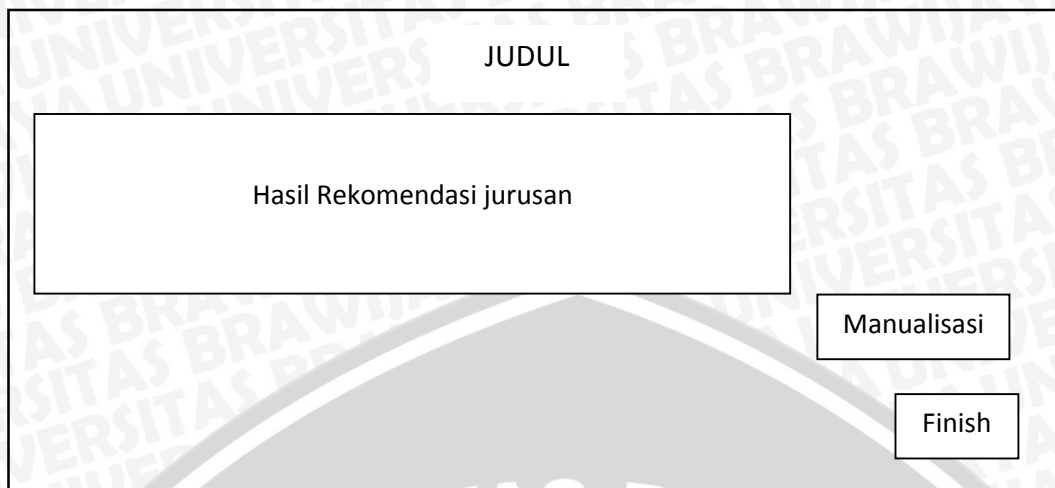
Perancangan manualisasi naive bayes menampilkan judul halaman, manualisasi perhitungan *naive bayes* untuk menampilkan probabilitas diterima calon mahasiswa di tiap fakultas, dan *button close* untuk menutup halaman manualisasi *naive bayes*. Perancangan manualisasi *naive bayes* dapat dilihat pada Gambar 4.5.



Gambar 4.5 Perancangan Halaman Manualisasi Naive Bayes

#### 4.1.6 Perancangan Testing AHP

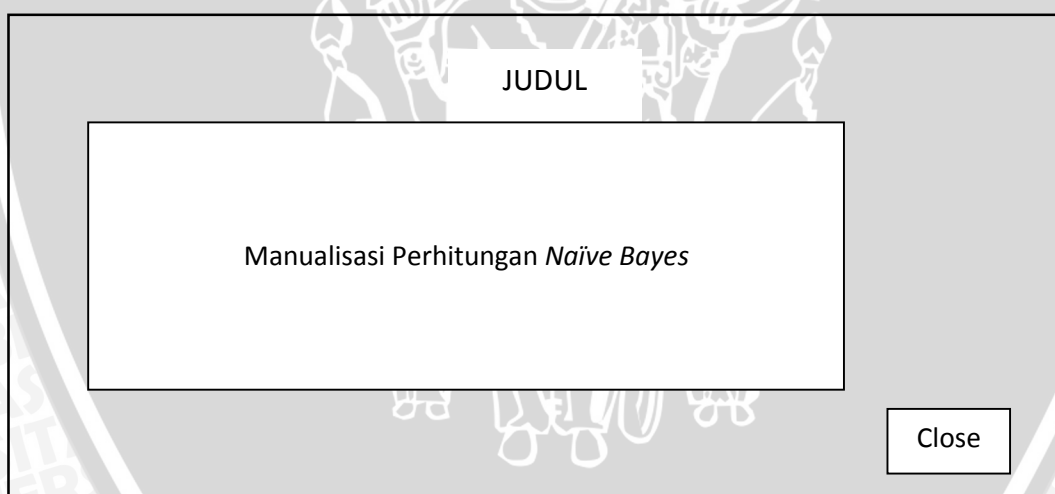
Perancangan *testing AHP* menampilkan judul halaman, Hasil Perhitungan *AHP* menampilkan hasil rekomendasi jurusan dari calon mahasiswa, *button manualisasi* menampilkan halaman manualisasi *AHP*, *button Finish* untuk menuju ke halaman menu utama. Perancangan *testing AHP* dapat dilihat pada Gambar 4.6.



Gambar 4.6 Perancangan Halaman Testing AHP

#### 4.1.7 Perancangan Manualisasi AHP

Perancangan manualisasi AHP menampilkan judul halaman, manualisasi metode AHP untuk menentukan hasil rekomendasi jurusan kuliah berdasarkan hasil klasifikasi fakultas metode *naïve bayes*, dan *button close* untuk menutup halaman manualisasi AHP. Perancangan manualisasi AHP dapat dilihat pada Gambar 4.7.



Gambar 4.7 Perancangan Halaman Manualisasi AHP

#### 4.2 Perancangan Uji Coba dan Evaluasi

Perancangan uji coba dan evaluasi disebabkan karena jumlah data sangat berpengaruh dalam metode *naïve bayes* dan juga matriks perbandingan berpasangan yang tercipta di AHP belum tentu konsisten. Oleh karena itu, untuk melakukan evaluasi sistem maka perlu dilakukan beberapa uji coba sebagai berikut :

1. Uji coba jumlah data latih algoritma Naïve Bayes
2. Uji coba jumlah data latih algoritma Naïve Bayes dengan data latih diskrit

3. Uji coba konsistensi hasil rekomendasi AHP

**4.2.1 Uji Coba Jumlah Data Latih Algoritma Naïve Bayes**

Uji coba banyaknya jumlah data latih algoritma *naïve bayes* digunakan untuk mendapatkan jumlah data latih yang optimal untuk tiap fakultas, untuk mendapatkan hasil klasifikasi yang baik. Rancangan uji coba untuk mengetahui jumlah data latih yang paling optimal dari tiap fakultas dapat dilihat pada Tabel 4.1.

**Tabel 4.1 Rancangan Uji Coba Jumlah Data Latih Algoritma Naïve Bayes**

Jumlah Data Latih tiap fakultas (persen)	Hasil Klasifikasi										Persentase Ketepatan
	1	2	3	4	5	6	7	8	9	10	
50											
70											
100											

**4.2.2 Uji Coba Jumlah Data Latih Algoritma Naïve Bayes dengan Data Latih Diskrit**

Uji coba banyaknya jumlah data latih algoritma *naïve bayes* dengan data latih diskrit memiliki kesamaan dengan uji coba jumlah data latih algoritma *naïve bayes*, perbedaannya adalah dengan mengubah terlebih dahulu data latih ke dalam bentuk diskrit. Tujuan uji coba ini adalah membandingkan akurasi sistem yang memiliki data latih normal dengan data latih diskrit. Rancangan uji coba jumlah data training algoritma *naïve bayes* dengan data training diskrit dapat dilihat pada Tabel 4.2.

**Tabel 4.2 Rancangan Uji Coba Jumlah Data Latih Algoritma Naïve Bayes dengan Data Latih Diskrit**

Jumlah Data Latih tiap fakultas (persen)	Hasil Klasifikasi										Persentase Ketepatan
	1	2	3	4	5	6	7	8	9	10	
50											
70											
100											



### 4.2.3 Uji Coba Konsistensi Hasil Rekomendasi AHP

Uji coba konsistensi hasil rekomendasi AHP terlebih dahulu menguji konsistensi logika matriks yang terbentuk melalui pembobotan nilai calon mahasiswa agar hasil rekomendasi yang didapat berasal dari matriks yang konsisten. Rancangan uji coba konsistensi matriks perbandingan bobot nilai algoritma AHP dapat dilihat pada Tabel 4.3.

**Tabel 4.3 Rancangan Uji Coba Konsistensi Matriks Perbandingan Bobot Nilai Algoritma AHP**

Percobaan ke- <i>i</i>	Nilai CR	Nilai CI	Hasil
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Setelah pengujian konsistensi matriks yang berasal dari bobot calon mahasiswa, dilakukan uji coba konsistensi hasil rekomendasi AHP merupakan uji coba untuk mengetahui tingkat konsisten sistem dalam memberikan rekomendasi dengan model perhitungan algoritma AHP yang berbeda. Terdapat 3 model yang akan dilakukan uji coba konsistensi hasil rekomendasi AHP ini. Model 1 AHP menggunakan semua mata pelajaran di dalam perhitungannya, model 2 AHP menggunakan semua mata pelajaran di dalam perhitungannya, khusus untuk mata pelajaran matematika, fisika, kimia, dan biologi dicari rata-ratanya kemudian diberi nama pelajaran sains, model 3 AHP hanya menggunakan mata pelajaran yang umum, yaitu Matematika, Fisika, Kimia, Biologi, Bahasa Indonesia, Bahasa Inggris, dan TIK. Rancangan uji coba untuk mengetahui tingkat konsistensi hasil rekomendasi AHP dapat dilihat pada Tabel 4.4.

**Tabel 4.4 Rancangan Uji Coba Konsistensi Hasil Rekomendasi AHP**

Data	Hasil Rekomendasi			Hasil Sebenarnya
	Model 1	Model 2	Model 3	
1				
2				
3				

4				
5				
6				
7				
8				
9				
10				



## BAB 5 IMPLEMENTASI

Pada bab ini akan dibahas mengenai implementasi dari perancangan sistem pengklasifikasi penentuan jurusan untuk pelajar SMA menggunakan metode AHP-Naïve Bayes. Bab ini terdiri dari spesifikasi sistem, batasan sistem, implementasi source code, dan implementasi user interface.

### 5.1 Spesifikasi Sistem

Spesifikasi sistem yang akan digunakan mengacu kepada spesifikasi sistem yang digunakan oleh penulis di dalam implementasinya. Spesifikasi sistem terbagi atas 2 bagian, yaitu spesifikasi perangkat lunak dan spesifikasi perangkat keras.

#### 5.1.1 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dijelaskan pada Tabel 5.1.

**Tabel 5.1 Spesifikasi Perangkat Lunak**

Nama Komponen	Spesifikasi
Sistem Operasi	Windows 8.1
Bahasa Pemrograman	Java
Tools pemrograman	Netbeans 8.1

#### 5.1.2 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dijelaskan pada Tabel 5.2.

**Tabel 5.2 Spesifikasi Perangkat Keras**

Nama Komponen	Spesifikasi
Processor	Windows 8.1
Memori (RAM)	4 GB
Storage	128 GB

#### 5.1.3 Batasan-Batasan Sistem

Terdapat beberapa batasan di dalam pengimplementasian sistem ini, yang dapat dijelaskan sebagai berikut :

1. *Input* kepada sistem adalah data nilai calon mahasiswa. Nilai yang dimasukkan berasal dari rata-rata nilai dari semester 3 hingga semester 5 calon pelajar tersebut.
2. *Output* sistem adalah rekomendasi fakultas beserta jurusan untuk calon mahasiswa.

3. Pengembangan sistem dilakukan dengan menggunakan bahasa pemrograman java.
4. Algoritma yang digunakan adalah Naïve Bayes dan AHP.

## 5.2 Implementasi Algoritma

Implementasi algoritma terbagi atas 2 proses utama, yaitu proses *Naïve Bayes* dan *Analytical Hierarchy Process*.

### 5.2.1 Implementasi Algoritma *Naïve Bayes*.

Implementasi algoritma *Naïve Bayes* terbagi atas beberapa bagian, yaitu :

- a. Proses Pendefinisian Data melalui File.

Di dalam program terdapat source code untuk mendefinisikan data berdasarkan isi file dataTesting.txt ke dalam program. Implementasi proses penginputan data melalui file dapat dilihat pada Gambar 5.1

```

1 File fileIn = new
2 File("../SistemRekomJurusan_2/dataTesting.txt");
3 try {
4     BufferedReader fIn = new BufferedReader(new
5     FileReader(fileIn));
6     fIn.mark(100000);
7
8     while (fIn.readLine() != null) {
9         banyakData++;
10    }
11
12    fIn.reset();
13
14    nilaiMhs = new String[banyakData][13];
15
16    while ((line = fIn.readLine()) != null) {
17        nilaiMhs[ind] = line.split("\t");
18        ind++;
19    }
20
21
22
23    } catch (FileNotFoundException ex) {
24
25    Logger.getLogger(naiveBayes.class.getName()).log(Level.S
26    EVERE, null, ex);
27
28    } catch (IOException ex) {
29
30    Logger.getLogger(naiveBayes.class.getName()).log(Level.S
31    EVERE, null, ex);
32    }
33

```

**Gambar 5.1** Pendefinisian Data Calon Mahasiswa

Penjelasan Gambar 5.1 adalah sebagai berikut :

Langkah awal adalah pendefinisian letak file dataTesting.txt . Kemudian dilakukan pembacaan file dengan menggunakan fungsi BufferedReader. Pembacaan file dilakukan untuk menentukan data nilai mahasiswa yang akan menjadi data uji. Data uji akan dimasukkan kedalam array nilaiMhs.

b. Proses Mendefinisikan Data Latih

Di dalam program terdapat source code untuk mendefinisikan data latih sesuai dengan daftar fakultas yang telah didefinisikan pada daftarFakultas.txt. Proses mendefinisikan data latih dapat dilihat pada Gambar 5.2

```

1 String dataNB[][] = null;
2     ind = 0;
3     int banyakfak = 0;
4     String data[] = null;
5     int totaldata[] = null;
6     String jurusan[] = null;
7     String total = "";
8
9     try {
10         BufferedReader baca = new
11         BufferedReader(new FileReader(fakultas));
12         String baris;
13         baca.mark(100000000);
14         while ((baca.readLine()) != null) {
15             baca.readLine();
16             banyakfak++;
17         }
18         data = new String[banyakfak];
19         totaldata = new int[banyakfak];
20         jurusan = new String[banyakfak];
21         for (int i = 0; i < jurusan.length; i++)
22         {
23             jurusan[i] = "";
24         }
25
26         baca.reset();
27         while ((baris = baca.readLine()) != null)
28         {
29             data[ind] = baris;
30             baris = baca.readLine();
31             String tempjur[] = baris.split(",");
32
33             for (int i = 0; i < tempjur.length;
34             i++) {
35                 jurusan[ind] += "tujuan = '" +
36                 tempjur[i] + "'";
37                 if (i < tempjur.length - 1) {
38                     jurusan[ind] += " or ";
39                 }
40             }
41             ind++;
42         }
43
44         String jurusansemua = "";
45

```

```

46         for (int i = 0; i < jurusan.length; i++)
47     {
48         jurusansemua += jurusan[i];
49         if (i < jurusan.length - 1) {
50             jurusansemua += " or ";
51         }
52     }
53
54     System.out.println(jurusansemua);
55
56     Connection c = koneksi.getKoneksi();
57
58     Statement s[] = new Statement[6];
59
60     for (int i = 0; i < s.length; i++) {
61         s[i] = c.createStatement();
62     }
63
64     String sql = "select * from
65     akademikpelajar where " + jurusansemua + " order by
66     hasil";
67     ResultSet r = s[0].executeQuery(sql);
68     String sql2 = "select found_rows() as
69     total";
70     ResultSet r2 = s[1].executeQuery(sql2);
71
72     while (r2.next()) {
73         total = r2.getString("total");
74     }
75
76     dataNB = new
77     String[Integer.parseInt(total)][17];
78
79
80     int i = 0;
81     while (r.next()) {
82
83         dataNB[i][0] = r.getString("no");
84         dataNB[i][1] = r.getString("agama");
85         dataNB[i][2] = r.getString("kwn");
86         dataNB[i][3] = r.getString("ind");
87         dataNB[i][4] = r.getString("ing");
88         dataNB[i][5] = r.getString("mat");
89         dataNB[i][6] = r.getString("fis");
90         dataNB[i][7] = r.getString("kim");
91         dataNB[i][8] = r.getString("bio");
92         dataNB[i][9] =
93     r.getString("sejarah");
94         dataNB[i][10] =
95     r.getString("senibudaya");
96         dataNB[i][11] =
97     r.getString("penjaskes");
98         dataNB[i][12] = r.getString("tik");
99         dataNB[i][13] = r.getString("kba");
100        dataNB[i][14] =
101     r.getString("tujuan");
102        dataNB[i][15] = r.getString("hasil");
103
104

```

```

105         for (int j = 0; j < jurusan.length;
106     j++) {
107             if
108     (jurusan[j].contains(dataNB[i][14])) {
109                 dataNB[i][16] = data[j];
110                 totaldata[j]++;
111             }
112         }
113
114         r.close();
115         r2.close();
116
117         for (int j = 0; j < s.length; j++) {
118             s[j].close();
119         }
120
121         } catch (SQLException ex) {
122
123     Logger.getLogger(settingUji.class.getName()).log(Level.S
124     EVERE, null, ex);
125         } catch (IOException ex) {
126
127     Logger.getLogger(settingUji.class.getName()).log(Level.S
128     EVERE, null, ex);
129         }

```

**Gambar 5.2 Pendefinisian Data Latih**

Penjelasan Gambar 5.2 adalah sebagai berikut :

Langkah awal adalah dengan menyiapkan variabel-variabel yang akan digunakan untuk proses pendefinisian data latih. Kemudian dilakukan proses pembacaan file berisi data fakultas. Data fakultas yang dibutuhkan adalah jumlah fakultas, nama dari masing-masing fakultas beserta jurusan yang terdapat di dalam fakultas tersebut. Setelah data fakultas didefinisikan, maka dilakukan proses eksekusi query untuk mengambil data latih dari database berdasarkan daftar jurusan yang telah didefinisikan.

#### c. Proses Perhitungan Hasil Naïve Bayes

Di dalam program terdapat source code untuk melakukan perhitungan hasil klasifikasi Naïve Bayes. Proses perhitungan hasil klasifikasi dilakukan dengan membandingkan prior total dari tiap fakultas. Fakultas yang memiliki nilai prior total tertinggi akan menjadi hasil klasifikasi. Proses perhitungan juga akan dituliskan kedalam file txt untuk menjadi acuan perhitungan berikutnya. Proses perhitungan hasil Naïve Bayes dapat dilihat pada Gambar 5.3

```

1     double max = -99;
2
3         try {
4             tulis.write("Manualisasi           Rekomendasi
5     Fakultas menggunakan metode Naive Bayes\n");
6             tulis.newLine();
7             tulis.newLine();
8             tulis.write("Nilai Calon Mahasiswa");
9             tulis.newLine();
10

```

```

11         for (int i = 0; i < nilaiMhs[1].length;
12         i++) {
13             tulis.write(nilaiMhs[1][i] + " ");
14         }
15         tulis.newLine();
16         tulis.newLine();
17
18         tulis.write("Hasil Perhitungan");
19         tulis.newLine();
20         tulis.write("Fakultas " + data[0]);
21         tulis.newLine();
22     } catch (IOException ex) {
23
24     Logger.getLogger(settingUji.class.getName()).log(Level.SEVERE,
25     null, ex);
26     }
27
28     double tidak[] = new double[14];
29
30     double        diterima[][] = new
31     double[data.length][14];
32
33     double totalya[] = new double[data.length];
34
35     double totalt = 0;
36
37     for (int i = 0; i < dataNB.length; i++) {
38         if
39         (dataNB[i][15].equalsIgnoreCase("diterima")) {
40             for (int j = 0; j < data.length; j++)
41             {
42                 if
43                 (dataNB[i][16].equalsIgnoreCase(data[j])) {
44                     totalya[j]++;
45                 }
46             }
47             } else {
48                 totalt++;
49             }
50         }
51
52         //hitung mean dan standar deviasi
53         double        mean[][] = new
54         double[jurusan.length][13];
55         double        sD[][] = new
56         double[jurusan.length][13];
57
58         double meantidak[] = new double[13];
59         double sDtidak[] = new double[13];
60
61         for (int i = 0; i < mean.length; i++) {
62             double totalmean = 0;
63             for (int j = 0; j < mean[0].length; j++) {
64                 totalmean = 0;
65                 for (int k = 0; k < dataNB.length; k++)
66                 {
67                     if
68                     ((jurusan[i].contains(dataNB[k][14])) &&
69                     dataNB[k][15].equalsIgnoreCase("diterima")) {

```



```

70         totalmean +=
71 Double.parseDouble(dataNB[k][j + 1]);
72     }
73     }
74     mean[i][j] = totalmean / totalya[i];
75 }
76 }
77
78     for (int i = 0; i < meantidak.length; i++) {
79         double totalmean = 0;
80         for (int k = 0; k < dataNB.length; k++) {
81             if
82 (dataNB[k][15].equalsIgnoreCase("tidak")) {
83                 totalmean +=
84 Double.parseDouble(dataNB[k][i + 1]);
85             }
86         }
87         meantidak[i] = totalmean / totalt;
88     }
89
90     for (int i = 0; i < sD.length; i++) {
91         double totalDataKurangMean = 0;
92         for (int j = 0; j < sD[0].length; j++) {
93             for (int k = 0; k < dataNB.length; k++)
94 {
95                 if
96 ((jurusan[i].contains(dataNB[k][14])) &&
97 dataNB[k][15].equalsIgnoreCase("diterima")) {
98                     totalDataKurangMean +=
99 Math.pow((Double.parseDouble(dataNB[k][j + 1]) -
100 mean[i][j]), 2);
101                 }
102             }
103             sD[i][j] =
104 Math.sqrt(totalDataKurangMean / (totalya[i] - 1));
105         }
106     }
107
108     for (int i = 0; i < sDtidak.length; i++) {
109         double totalDataKurangMean = 0;
110         for (int j = 0; j < dataNB.length; j++) {
111             if
112 (dataNB[j][15].equalsIgnoreCase("tidak")) {
113                 totalDataKurangMean +=
114 Math.pow((Double.parseDouble(dataNB[j][i + 1]) -
115 meantidak[i]), 2);
116             }
117         }
118         sDtidak[i] = Math.sqrt(totalDataKurangMean
119 / (totalt - 1));
120     }
121
122 //     hitung prior
123     for (int i = 0; i < data.length; i++) {
124         diterima[i][0] = totalya[i] /
125 Integer.parseInt(total);
126     }
127
128

```

```

129         tidak[0] = (double) (totalt) /
130 Integer.parseInt(total);
131
132         for (int i = 0; i < data.length; i++) {
133             System.out.print(totalya[i] + "/" +
134 Integer.parseInt(total) + "\t");
135         }
136
137         System.out.println("\t" + totalt + "/" +
138 Integer.parseInt(total));
139
140         int id = 1;
141
142         try {
143             tulis.write("Prior kelas");
144             for (int j = 0; j < data.length; j++) {
145                 tulis.write("diterima " + data[j] + "
146 : " + diterima[j][0] + "\t");
147             }
148             tulis.write("\ttidak diterima : " +
149 tidak[0]);
150             tulis.newLine();
151         } catch (IOException ex) {
152
153 Logger.getLogger(settingUji.class.getName()).log(Level.SEVERE, null, ex);
154
155         }
156
157
158         for (int i = 0; i < diterima[0].length - 1;
159 i++) {
160             //
161
162             try {
163
164                 tulis.write("LikeLikhood tiap mata
165 pelajaran");
166                 tulis.newLine();
167                 tulis.write("Mata pelajaran " + i);
168                 tulis.newLine();
169                 for (int j = 0; j < data.length; j++)
170                 {
171                     diterima[j][i + 1] = (Double) (1 /
172 Math.sqrt(2 * Math.PI * sD[j][i])) * Math.pow(Math.E, -
173 (Math.pow((nilaiAngka[i] - mean[j][i]), 2) / (2 *
174 Math.pow(sD[j][i], 2))));
175                     tulis.write("diterima " + data[j]
176 + " : " + diterima[j][i] + "\t");
177                 }
178
179                 tidak[i + 1] = (Double) (1 /
180 Math.sqrt(2 * Math.PI * sDtidak[i])) * Math.pow(Math.E, -
181 (Math.pow((nilaiAngka[i] - meantidak[i]), 2) / (2 *
182 Math.pow(sDtidak[i], 2))));
183                 tulis.write("\ttidak diterima : " +
184 tidak[i]);
185
186                 tulis.newLine();
187             } catch (IOException ex) {

```

```
188
189  Logger.getLogger(settingUji.class.getName()).log(Level.SEVERE, null, ex);
190
191  }
192
193  }
194
195  double totalditerima[] = new
196  double[data.length];
197  for (int i = 0; i < totalditerima.length; i++)
198  {
199      totalditerima[i] = 1;
200  }
201
202  double totaltidak = 1.0;
203
204  for (int i = 0; i < diterima[0].length; i++) {
205      for (int j = 0; j < data.length; j++) {
206          totalditerima[j] *= diterima[j][i];
207      }
208      totaltidak *= tidak[i];
209  }
210
211  try {
212      tulis.write("Posterior");
213      tulis.newLine();
214      for (int i = 0; i < data.length; i++) {
215          tulis.write("diterima " + data[i] + "
216 : " + totalditerima[i] + "\t");
217      }
218      tulis.write("tidak diterima : " +
219 totaltidak);
220      tulis.newLine();
221      tulis.newLine();
222  } catch (IOException ex) {
223
224  Logger.getLogger(settingUji.class.getName()).log(Level.SEVERE, null, ex);
225
226  }
227
228  for (int i = 0; i < data.length; i++) {
229      System.out.printf("%.10f",
230 totalditerima[i]);
231  }
232
233  System.out.printf("%.10f", totaltidak);
234  System.out.println("");
235
236  String fakultaspil = data[0];
237  max = totalditerima[0];
238  for (int i = 1; i < data.length; i++) {
239      if (max < totalditerima[i]) {
240          max = totalditerima[i];
241          fakultaspil = data[i];
242      }
243  }
244
245  if (max < totaltidak) {
246      max = totaltidak;
```

```

247         fakultaspil = "tidak";
248     }
249
250     System.out.println("fakultas      : " +
251 fakultaspil);
252
253     try {
254         tulis.write(fakultaspil);
255         tulis.newLine();
256         tulis.newLine();
257         tulisHasil.write(fakultaspil);
258         tulisHasil.newLine();
259
260         if
261 (!fakultaspil.equalsIgnoreCase("tidak")) {
262             for (int i = 0; i < nilaiMhs[0].length;
263 i++) {
264                 try {
265
266 tulisData.write(nilaiMhs[l][i] + " ");
267                     } catch (IOException ex) {
268
269 Logger.getLogger(settingUji.class.getName()).log(Level.SEVERE, null, ex);
270                 }
271             }
272         }
273         tulisData.write(fakultaspil);
274         tulisData.newLine();
275     }
276     } catch (IOException ex) {
277
278     }
279     }
280
281     try {
282         tulis.close();
283         tulisHasil.close();
284         tulisData.close();
285     } catch (IOException ex) {
286
287 Logger.getLogger(settingUji.class.getName()).log(Level.SEVERE, null, ex);
288     }
289     }

```

**Gambar 5.3 Perhitungan hasil Naïve Bayes**

Penjelasan Gambar 5.3 adalah sebagai berikut :

Proses ini membutuhkan variabel max untuk menentukan indeks fakultas dengan nilai posterior tertinggi. Nilai awal variabel max diatur -99 dengan asumsi fakultas pertama memiliki indeks 0. Kemudian dilakukan proses perhitungan metode naïve bayes sekaligus mencatat prosesnya ke dalam file manualisasi untuk ditampilkan. Sebelum proses dilakukan, dilakukan inisialisasi data array untuk menampung nilai prior untuk masing-masing kemungkinan fakultas, kemudian dilakukan proses perhitungan prior untuk masing-masing fakultas untuk mendapatkan nilai posterior. Nilai prior dihitung dengan menentukan peluang fakultas tersebut terpilih dengan mempertimbangkan syarat yaitu nilai tiap mata

pelajaran. Setelah nilai prior ditentukan, nilai posterior didapatkan dengan mengalikan semua nilai prior untuk masing-masing fakultas. fakultas yang memiliki nilai posterior tertinggi akan menjadi hasil klasifikasi dari data uji yang telah dimasukkan. Kemudian proses dilakukan kembali sebanyak jumlah data uji yang diinputkan. Data uji yang memiliki hasil selain tidak akan dituliskan pada file dataAHP.txt untuk dilakukan proses perhitungan menggunakan metode AHP.

### 5.2.2 Implementasi Algoritma AHP.

Implementasi algoritma AHP dilakukan dengan 3 model. Implementasi algoritma AHP terbagi atas beberapa bagian untuk masing-masing model, yaitu :

a. Proses pengambilan data nilai calon mahasiswa

Di dalam program terdapat source code untuk mengambil data dari file manualisasi naïve bayes. Implementasi dapat dilihat pada Gambar 5.4

```

1 File fileIn = new
2 File("../SistemRekomJurusan_2/dataAHP.txt");
3 try {
4     BufferedReader fIn = new BufferedReader(new
5     FileReader(fileIn));
6     fIn.mark(1000);
7
8     while ((line = fIn.readLine()) != null) {
9         totaldata++;
10    }
11
12    nilaiMhs = new String[totaldata][14];
13
14    fIn.reset();
15
16    while ((line = fIn.readLine()) != null) {
17        nilaiMhs[ind] = line.split(" ");
18        ind++;
19    }
20
21    } catch (FileNotFoundException ex) {
22
23    Logger.getLogger(naiveBayes.class.getName()).log(Level.SEVERE,
24    null, ex);
25
26    } catch (IOException ex) {
27
28    Logger.getLogger(naiveBayes.class.getName()).log(Level.SEVERE,
29    null, ex);
30    }

```

**Gambar 5.4 Pengambilan Data Nilai Calon Mahasiswa**

Penjelasan dari Gambar 5.4 adalah sebagai berikut :

Langkah awal adalah mendapatkan data uji yang berasal dari data klasifikasi naïve bayes yang memiliki hasil klasifikasi selain tidak. Proses mendapatkan data dilakukan dengan membaca file dataAHP.txt menggunakan fungsi `BufferedReader`. Data yang terbaca kemudian akan dimasukkan ke dalam array `nilaiMhs`.

- b. Proses pengambilan data nilai minimum untuk setiap mata pelajaran untuk model 1.

Di dalam program terdapat source code untuk mendapatkan nilai minimum untuk setiap mata pelajaran untuk setiap fakultas. Data nilai minimum didapatkan dari database. Implementasi dapat dilihat pada Gambar 5.5

```

1 String minFakultas[][] = null;
2
3     int jumlahJurusan = 0;
4
5     try {
6
7         Connection c = koneksi.getKoneksi();
8
9         Statement[] s = new Statement[2];
10
11        for (int i = 0; i < s.length; i++) {
12            s[i] = c.createStatement();
13        }
14
15        ResultSet r[] = new ResultSet[2];
16
17        ind = 0;
18
19        String sql = "select tujuan,min(agama) as
20 minagama,min(kwn),min(ind),min(ing),min(mat),min(fis) "
21 +
22 ",min(kim),min(bio),min(sejarah),min(senibudaya),min(penj
23 askes),min(tik) "
24 + ",min(kba) from akademikpelajar
25 where (" + daftarjurusan[penanda] + ") and hasil='diterima'
26 group by tujuan";
27        r[0] = s[0].executeQuery(sql);
28        String sql2 = "select found_rows() as
29 jumlahbaris";
30        r[1] = s[1].executeQuery(sql2);
31
32        while (r[1].next()) {
33            jumlahJurusan =
34 Integer.parseInt(r[1].getString("jumlahbaris"));
35        }
36
37        minFakultas = new
38 String[jumlahJurusan][14];
39
40        daftarJurusanUrut = new
41 String[jumlahJurusan];
42
43        ind = 0;
44
45        int i = 0;
46        while (r[0].next()) {
47            minFakultas[i][0] =
48 r[0].getString("tujuan");
49            daftarJurusanUrut[ind] =
50 r[0].getString("tujuan");
51

```

```

52         minFakultas[i][1] =
53         r[0].getString("minagama");
54         minFakultas[i][2] =
55         r[0].getString("min(kwn)");
56         minFakultas[i][3] =
57         r[0].getString("min(ind)");
58         minFakultas[i][4] =
59         r[0].getString("min(ing)");
60         minFakultas[i][5] =
61         r[0].getString("min(mat)");
62         minFakultas[i][6] =
63         r[0].getString("min(fis)");
64         minFakultas[i][7] =
65         r[0].getString("min(kim)");
66         minFakultas[i][8] =
67         r[0].getString("min(bio)");
68         minFakultas[i][9] =
69         r[0].getString("min(sejarah)");
70         minFakultas[i][10] =
71         r[0].getString("min(senibudaya)");
72         minFakultas[i][11] =
73         r[0].getString("min(penjaskes)");
74         minFakultas[i][12] =
75         r[0].getString("min(tik)");
76         minFakultas[i][13] =
77         r[0].getString("min(kba)");
78         ind++;
79         i++;
80     }
81
82     r[0].close();
83     r[1].close();
84
85     //
86     s[0].close();
87     s[1].close();
88
89     } catch (SQLException ex) {
90
91     Logger.getLogger(settingUji.class.getName()).log(Level.SEVERE, null, ex);
92
93     }

```

**Gambar 5.5 Pengambilan Data Nilai Minimum Tiap Fakultas**

Penjelasan Gambar 5.5 adalah sebagai berikut :

Langkah awal adalah inialisasi variabel array minFakultas untuk menampung nilai minimum tiap mata pelajaran untuk fakultas yang diproses. Fakultas yang diproses ditentukan melalui hasil klasifikasi data yang didapat dari algoritma naïve bayes. Kemudian dilakukan proses untuk mengeksekusi query yang berisi perintah untuk mendapatkan nilai minimum tiap mata pelajaran berdasarkan dari data latih yang memiliki hasil diterima. Kemudian dilakukan pemasukan data hasil query ke dalam array minFakultas untuk proses perhitungan selanjutnya.

## c. Proses pembuatan matriks nilai calon mahasiswa

Di dalam program terdapat source code untuk membuat matriks perbandingan berpasangan berdasarkan nilai dari calon mahasiswa. Implementasi dapat dilihat pada Gambar 5.6

```

1  tulis.write("Bobot nilai Mahasiswa");
2  tulis.newLine();
3  double bobotNilai[] = new
4  double[nilaiAngka.length];
5  for (int j = 0; j < bobotNilai.length; j++)
6  {
7      if (nilaiAngka[j] < 75) {
8          bobotNilai[j] = 1;
9      } else if (nilaiAngka[j] >= 75 &&
10 nilaiAngka[j] <= 80) {
11         bobotNilai[j] = 2;
12     } else if (nilaiAngka[j] > 80 &&
13 nilaiAngka[j] <= 85) {
14         bobotNilai[j] = 3;
15     } else if (nilaiAngka[j] > 85 &&
16 nilaiAngka[j] <= 90) {
17         bobotNilai[j] = 4;
18     } else if (nilaiAngka[j] > 90) {
19         bobotNilai[j] = 5;
20     }
21 }
22
23 double matriksBobot[][] = new
24 double[nilaiAngka.length][nilaiAngka.length];
25 double totalbobot1[] = new
26 double[nilaiAngka.length];
27 double normalisasi[][] = new
28 double[nilaiAngka.length][nilaiAngka.length];
29 double totalnorm[] = new
30 double[nilaiAngka.length];
31 double rata[] = new
32 double[nilaiAngka.length];
33 double lambda[] = new
34 double[nilaiAngka.length];
35 double totalLambda = 0;
36 double lambdaMax = 0;
37 double CI = 0;
38 double RI = 1.56;
39 double CR = 0;
40
41 for (int j = 0; j < nilaiAngka.length; j++)
42 {
43     for (int k = 0; k < (nilaiAngka.length);
44 k++) {
45         double perb = 0;
46         if (j == k) {
47             perb = 1;
48         } else if (Math.abs(bobotNilai[j] -
49 bobotNilai[k]) == 1) {
50             perb = 3;
51         }
52     }

```



```

53         } else if (Math.abs(bobotNilai[j] -
54 bobotNilai[k]) == 2) {
55             perb = 5;
56         } else if (Math.abs(bobotNilai[j] -
57 bobotNilai[k]) == 3) {
58             perb = 7;
59         } else if (Math.abs(bobotNilai[j] -
60 bobotNilai[k]) == 4) {
61             perb = 9;
62         } else if (Math.abs(bobotNilai[j] -
63 bobotNilai[k]) == 0) {
64             perb = 1;
65         }
66
67         if (bobotNilai[j] < bobotNilai[k]) {
68             perb = 1 / perb;
69         }
70
71         matriksBobot[j][k] = perb;
72         totalbobot1[k] += perb;
73     }
74 }
75
76 for (int j = 0; j < nilaiAngka.length; j++)
77 {
78     for (int k = 0; k < nilaiAngka.length;
79 k++) {
80         normalisasi[j][k] =
81 matriksBobot[j][k] / Math.round(totalbobot1[k]);
82         totalnorm[j] += normalisasi[j][k];
83     }
84 }
85
86 for (int i = 0; i < rata.length; i++) {
87     rata[i] = totalnorm[i] /
88 nilaiAngka.length;
89 }
90
91 for (int i = 0; i < lambda.length; i++) {
92     lambda[i] = totalnorm[i] / rata[i];
93     tulis.write(totalnorm[i] + " : " +
94 rata[i]);
95     tulis.newLine();
96     tulis.write("lambda " + i + " : " +
97 lambda[i]);
98     tulis.newLine();
99     totalLambda += lambda[i];
100 }
101
102 tulis.write("Total lambda : " +
103 totalLambda);
104 tulis.newLine();
105 lambdaMax = totalLambda /
106 Double.valueOf(lambda.length);
107
108 tulis.write("lambdaMax : " + lambdaMax);

```

```

112     tulis.newLine();
113     CI = (lambdaMax -
114 Double.valueOf(lambda.length)
115 Double.valueOf(lambda.length - 1);
116     tulis.write("CI : " + CI);
117     tulis.newLine();
118     CR = CI / RI;
119
120     tulis.write("CR : " + CR);
121     tulis.newLine();
122
123     if (CR < 0.1) {
124         tulis.write("Konsisten");
125         tulis.newLine();
126     } else {
127         tulis.write("Tidak Konsisten");
128         tulis.newLine();
129     }

```

**Gambar 5.6 Pembuatan Matriks Nilai Calon Mahasiswa**

Penjelasan Gambar 5.6 adalah sebagai berikut :

Langkah awal adalah inialisasi variabel array bobotNilai untuk menampung nilai bobot dari nilai mata pelajaran data uji. Pemberian bobot memiliki aturan yang dapat dilihat pada Tabel 3.7. Jika nilai kurang dari 75 akan memiliki bobot 1, jika nilai berada di antara nilai 75 hingga 80, maka akan memiliki bobot 2, jika nilai berada di antara 81 hingga 85 akan memiliki bobot 3, jika nilai berada di antara nilai 86 hingga 90 akan memiliki bobot 4, dan jika nilai lebih dari 90, maka akan memiliki bobot 5. Kemudian dilakukan proses inialisasi variabel yang berfungsi untuk uji konsistensi matriks perbandingan berpasangan. Matriks perbandingan berpasangan dibuat dengan membandingkan nilai bobot dari masing-masing mata pelajarannya secara berpasangan, dengan aturan bervariasi untuk tiap model AHP. Matriks yang terbentuk kemudian akan diuji konsistensinya dengan membuat matriks normalisasi serta menghitung rata-rata dan lambda. Lambda berguna untuk menghitung CI yang nantinya akan digunakan dalam perhitungan CR untuk menentukan tingkat konsistensi matriks. Jika nilai CR berada di bawah 0.1, maka matriks dinyatakan konsisten.

#### d. Pembobotan Mata Pelajaran untuk Tiap Fakultas

Di dalam program terdapat source code untuk memberikan bobot kepada tiap mata pelajaran pada suatu fakultas. Pemberian bobot dilakukan dengan membandingkan nilai minimum mata pelajaran dengan nilai calon mahasiswa pada mata pelajaran yang sama. Implementasi dapat dilihat pada Gambar 5.7

```

1     System.out.println("temp nilai");
2     String tempNilai[][] = new
3     String[minTeknik.length][minTeknik[0].length - 1];
4
5     for (int j = 0; j < tempNilai.length; j++) {
6         for (int k = 0; k < tempNilai[0].length;
7         k++) {
8             tempNilai[j][k] = minTeknik[j][k +
9             1];

```

```
10     }
11     }
12
13     for (int j = 0; j < tempNilai.length; j++) {
14         for (int k = 0; k < tempNilai[0].length;
15 k++) {
16             System.out.print(tempNilai[j][k] + "
17 ");
18         }
19         System.out.println("");
20     }
21 //
22
23     //String bobot[][] = minTeknik;
24     for (int j = 0; j < tempNilai[0].length; j++)
25 {
26         for (int l = 0; l < tempNilai.length -
27 2; l++) {
28             for (int k = 0; k < tempNilai.length
29 - 1; k++) {
30                 if
31 ((Double.parseDouble(tempNilai[k][j])) >
32 (Double.parseDouble(tempNilai[k + 1][j]))) {
33                     String temp =
34 tempNilai[k][j];
35                     tempNilai[k][j] =
36 tempNilai[k + 1][j];
37                     tempNilai[k + 1][j] = temp;
38                 }
39             }
40         }
41     }
42
43     for (int j = 0; j < tempNilai.length; j++) {
44         for (int k = 0; k < tempNilai[0].length;
45 k++) {
46             System.out.print(tempNilai[j][k] + "
47 ");
48         }
49         System.out.println("");
50     }
51
52     //matriks untuk menentukan posisi nilai
53 mahasiswa di di urutan jurusan
54     System.out.println("posisi nilai
55 mahasiswa");
56     int urutanNilai[] = new
57 int[tempNilai[0].length];
58
59     for (int j = 0; j < tempNilai[0].length; j++)
60 {
61         int isi = 0;
62         for (int k = 0; k < tempNilai.length;
63 k++) {
64             if (nilaiAngka[j] >
65 Double.parseDouble(tempNilai[k][j])) {
66                 isi++;
67             } else {
68                 break;
```

```
69         }
70     }
71     urutanNilai[j] = isi;
72 }
73
74 for (int j = 0; j < nilaiAngka.length; j++)
75 {
76     System.out.print(nilaiAngka[j] + " ");
77 }
78 System.out.println("");
79 System.out.println("-----");
80 ");
81     for (int j = 0; j < urutanNilai.length; j++)
82 {
83     System.out.print(urutanNilai[j] + " ");
84 }
85
86 //matrik pemberian bobot tanpa terurut
87 System.out.println("bobot tidak urut");
88 String bobotMatpel[][] = new
89 String[tempNilai.length][tempNilai[0].length];
90
91     for (int j = 0; j < tempNilai[0].length; j++)
92 {
93     int bobotN = 1;
94     for (int k = urutanNilai[j]; k <
95 tempNilai.length; k++) {
96         bobotMatpel[k][j] = "1";
97     }
98     for (int k = 0; k < urutanNilai[j]; k++)
99 {
100
101         bobotMatpel[k][j] =
102 String.valueOf(bobotN);
103         bobotN++;
104     }
105 }
106
107     for (int j = 0; j < bobotMatpel.length; j++)
108 {
109     for (int k = 0; k <
110 bobotMatpel[0].length; k++) {
111         System.out.print(bobotMatpel[j][k]
112 + " ");
113     }
114     System.out.println("");
115 }
116
117 //bobot tiap jurusan dengan pembobotan urut
118 System.out.println("Bobot urut");
119 String bobotReal[][] = new
120 String[bobotMatpel.length][bobotMatpel[0].length];
121
122     for (int j = 0; j < bobotReal[0].length; j++)
123 {
124     for (int k = 0; k < bobotReal.length;
125 k++) {
126     for (int l = 0; l < bobotReal.length;
127 l++) {
```

```

128 //System.out.println(minTeknik[k][j+1]+" dan
129 "+tempNilai[l][k]);
130
131
132         if      (minTeknik[k][j      +
133 1].equalsIgnoreCase(tempNilai[l][j])) {
134
135             bobotReal[k][j] =
136 bobotMatpel[l][j];
137
138         }
139     }
140
141     }
142 }
143
144     System.out.println("-----
145 -");
146     for (int j = 0; j < bobotReal.length; j++) {
147         for (int k = 0; k < bobotReal[0].length;
148 k++) {
149             System.out.print(bobotReal[j][k] + "
150 ");
151         }
152     System.out.println("");
153 }

```

**Gambar 5.7 Pembobotan Mata Pelajaran**

Penjelasan Gambar 5.7 adalah sebagai berikut :

Langkah awal adalah membuat variabel array tempNilai untuk menentukan bobot dari masing-masing mata pelajaran untuk tiap fakultas. variabel tempNilai akan diisi dengan data nilai minimum mata pelajaran tiap fakultas, kemudian data nilai minimum tiap mata pelajaran akan diurutkan berdasarkan mata pelajarannya. Kemudian dilakukan pencarian urutan nilai data uji terhadap urutan data nilai minimum tersebut. Jika nilai data uji lebih tinggi dari nilai minimum di jurusan tertentu, maka bobot jurusan tersebut akan bertambah sesuai urutan kedekatan nilai data uji dengan nilai data minimum di jurusan itu. Jika nilai data uji lebih kecil dari nilai minimum jurusan tertentu, maka jurusan memiliki bobot 1 untuk mata pelajaran tersebut. Proses ini dilakukan hingga semua mata pelajaran di tiap jurusan mendapatkan bobot.

e. Membuat Matriks Perbandingan Berpasangan Untuk Setiap Mata Pelajaran

Di dalam program terdapat source code untuk membuat matriks perbandingan berpasangan untuk setiap mata pelajaran di fakultas tertentu. Pembuatan matriks perbandingan berpasangan akan dibuat dengan 3 model sesuai dengan variasi model AHP. Perbedaan tiap model adalah mata pelajaran yang diperhitungkan untuk menentukan hasil rekomendasi. Implementasi untuk model 1 dapat dilihat pada Gambar 5.8

```

1 double[][] matrikAgama = new
2 double[jumlahJurusan][jumlahJurusan];
3 double[][] matrikKwn = new
4 double[jumlahJurusan][jumlahJurusan];

```



```

5         double[][]      matrikInd      =      new
6 double[jumlahJurusan][jumlahJurusan];
7         double[][]      matrikIng      =      new
8 double[jumlahJurusan][jumlahJurusan];
9         double[][]      matrikMat      =      new
10 double[jumlahJurusan][jumlahJurusan];
11        double[][]      matrikFis      =      new
12 double[jumlahJurusan][jumlahJurusan];
13        double[][]      matrikKim      =      new
14 double[jumlahJurusan][jumlahJurusan];
15        double[][]      matrikBio      =      new
16 double[jumlahJurusan][jumlahJurusan];
17        double[][]      matrikSjr      =      new
18 double[jumlahJurusan][jumlahJurusan];
19        double[][]      matrikSnb      =      new
20 double[jumlahJurusan][jumlahJurusan];
21        double[][]      matrikPenjas   =      new
22 double[jumlahJurusan][jumlahJurusan];
23        double[][]      matrikTIK      =      new
24 double[jumlahJurusan][jumlahJurusan];
25        double[][]      matrikKba      =      new
26 double[jumlahJurusan][jumlahJurusan];
27
28        double[][]      totalbobot     =      new
29 double[13][jumlahJurusan];
30        double[][]      totalbobotnorm =      new
31 double[13][jumlahJurusan];
32        double[][]      ratabobot      =      new
33 double[jumlahJurusan][13];
34
35        for (int j = 0; j < jumlahJurusan; j++) {
36            for (int k = 0; k < jumlahJurusan; k++)
37            {
38                double bobotmapel = 0;
39                if
40 (Math.abs(Double.parseDouble(bobotReal[j][0])
41 Double.parseDouble(bobotReal[k][0])) == 0) {
42                    bobotmapel = 1;
43                } else if
44 (Math.abs(Double.parseDouble(bobotReal[j][0])
45 Double.parseDouble(bobotReal[k][0])) <= 2) {
46                    bobotmapel = 3;
47                } else if
48 (Math.abs(Double.parseDouble(bobotReal[j][0])
49 Double.parseDouble(bobotReal[k][0])) <= 4) {
50                    bobotmapel = 5;
51                } else if
52 (Math.abs(Double.parseDouble(bobotReal[j][0])
53 Double.parseDouble(bobotReal[k][0])) <= 6) {
54                    bobotmapel = 7;
55                } else if
56 (Math.abs(Double.parseDouble(bobotReal[j][0])
57 Double.parseDouble(bobotReal[k][0])) <= 10) {
58                    bobotmapel = 9;
59                }
60
61                if
62 (Double.parseDouble(bobotReal[j][0])
63 Double.parseDouble(bobotReal[k][0])) {

```

```

64         bobotmapel = 1 / bobotmapel;
65     }
66     matrikAgama[j][k] = bobotmapel;
67     //tulis.write(matrikAgama[j][k] + "
68 ");
69     totalbobot[0][k] += bobotmapel;
70 }
71 //tulis.newLine();
72 }
73 for (int j = 0; j < jumlahJurusan; j++) {
74     for (int k = 0; k < jumlahJurusan; k++)
75     {
76         matrikAgama[j][k] =
77         matrikAgama[j][k] / totalbobot[0][k];
78         tulis.write(matrikAgama[j][k] + "
79 ");
80         totalbobotnorm[0][j] +=
81         matrikAgama[j][k];
82     }
83     tulis.newLine();
84 }
85 for (int j = 0; j < matrikAgama.length; j++)
86 {
87     tulis.write(totalbobotnorm[0][j] + "
88 ");
89     tulis.newLine();
90     for (int j = 0; j < jumlahJurusan; j++) {
91         ratabobot[j][0] = totalbobotnorm[0][j]
92         / jumlahJurusan;
93         //tulis.write(totalbobot[0][j]+ " ");
94     }
95     tulis.newLine();
96     for (int j = 0; j < jumlahJurusan; j++) {
97         //bobot KWN
98         tulis.write("Kwn");
99         tulis.newLine();
100        for (int j = 0; j < jumlahJurusan; j++) {
101            for (int k = 0; k < jumlahJurusan; k++)
102            {
103                double bobotmapel = 0;
104                if
105                (Math.abs(Double.parseDouble(bobotReal[j][1]) -
106                Double.parseDouble(bobotReal[k][1])) == 0) {
107                    bobotmapel = 1;
108                } else if
109                (Math.abs(Double.parseDouble(bobotReal[j][1]) -
110                Double.parseDouble(bobotReal[k][1])) <= 2) {
111                    bobotmapel = 3;
112                }
113            }
114        }
115    }
116 }
117 }
118 }
119 }
120 }
121 }
122 }

```

```

123         } else if
124 (Math.abs(Double.parseDouble(bobotReal[j][1])
125 Double.parseDouble(bobotReal[k][1])) <= 4) {
126         bobotmapel = 5;
127     } else if
128 (Math.abs(Double.parseDouble(bobotReal[j][1])
129 Double.parseDouble(bobotReal[k][1])) <= 6) {
130         bobotmapel = 7;
131     } else if
132 (Math.abs(Double.parseDouble(bobotReal[j][1])
133 Double.parseDouble(bobotReal[k][1])) <= 10) {
134         bobotmapel = 9;
135     }
136
137     if
138 (Double.parseDouble(bobotReal[j][1]) <
139 Double.parseDouble(bobotReal[k][1])) {
140         bobotmapel = 1 / bobotmapel;
141     }
142
143     matrikKwn[j][k] = bobotmapel;
144     //tuliskan.write(matrikKwn[j][k]+" ");
145     totalbobot[1][k] += bobotmapel;
146 }
147
148 // ratabobot[j][1] = totalbobot[1][j]
149 / jumlahJurusan;
150 // tuliskan.newLine();
151 }
152 for (int j = 0; j < jumlahJurusan; j++) {
153     for (int k = 0; k < jumlahJurusan; k++)
154 {
155         matrikKwn[j][k] = matrikKwn[j][k] /
156 totalbobot[1][k];
157         tuliskan.write(matrikKwn[j][k] + " ");
158         totalbobotnorm[1][j] +=
159 matrikKwn[j][k];
160     }
161     tuliskan.newLine();
162 }
163
164 for (int j = 0; j < jumlahJurusan; j++) {
165     ratabobot[j][1] = totalbobotnorm[1][j]
166 / jumlahJurusan;
167 }
168
169     tuliskan.newLine();
170
171     //bobotInd
172     tuliskan.write("Bind");
173     tuliskan.newLine();
174     for (int j = 0; j < jumlahJurusan; j++) {
175         for (int k = 0; k < jumlahJurusan; k++)
176 {
177             double bobotmapel = 0;
178             if
179 (Math.abs(Double.parseDouble(bobotReal[j][2])
180 Double.parseDouble(bobotReal[k][2])) == 0) {
181                 bobotmapel = 1;

```



```

182         } else if
183 (Math.abs(Double.parseDouble(bobotReal[j][2])
184 Double.parseDouble(bobotReal[k][2])) <= 2) {
185     bobotmapel = 3;
186     } else if
187 (Math.abs(Double.parseDouble(bobotReal[j][2])
188 Double.parseDouble(bobotReal[k][2])) <= 4) {
189     bobotmapel = 5;
190     } else if
191 (Math.abs(Double.parseDouble(bobotReal[j][2])
192 Double.parseDouble(bobotReal[k][2])) <= 6) {
193     bobotmapel = 7;
194     } else if
195 (Math.abs(Double.parseDouble(bobotReal[j][2])
196 Double.parseDouble(bobotReal[k][2])) <= 10) {
197     bobotmapel = 9;
198     }
199
200     if
201 (Double.parseDouble(bobotReal[j][2]) <
202 Double.parseDouble(bobotReal[k][2])) {
203     bobotmapel = 1 / bobotmapel;
204     }
205
206     matrikInd[j][k] = bobotmapel;
207     //tuliskan.write(matrikInd[j][k]+" ");
208     totalbobot[2][k] += bobotmapel;
209     }
210
211     //ratabobot[j][2] = totalbobot[2][j] /
212 jumlahJurusan;
213     //tuliskan.newLine();;
214     }
215     for (int j = 0; j < jumlahJurusan; j++) {
216     for (int k = 0; k < jumlahJurusan; k++)
217     {
218     matrikInd[j][k] = matrikInd[j][k] /
219 totalbobot[2][k];
220     tuliskan.write(matrikInd[j][k] + " ");
221     totalbobotnorm[2][k] +=
222 matrikInd[j][k];
223     }
224     tuliskan.newLine();
225     }
226
227     for (int j = 0; j < jumlahJurusan; j++) {
228     ratabobot[j][2] = totalbobotnorm[2][j]
229 / jumlahJurusan;
230
231     }
232     //tuliskan.newLine();;
233
234     tuliskan.newLine();
235
236     //bobot Ing
237     tuliskan.write("Bing");
238     tuliskan.newLine();
239     for (int j = 0; j < jumlahJurusan; j++) {
240

```

```

241         for (int k = 0; k < jumlahJurusan; k++)
242     {
243         double bobotmapel = 0;
244         if
245     (Math.abs(Double.parseDouble(bobotReal[j][3]) -
246     Double.parseDouble(bobotReal[k][3])) == 0) {
247             bobotmapel = 1;
248         } else if
249     (Math.abs(Double.parseDouble(bobotReal[j][3]) -
250     Double.parseDouble(bobotReal[k][3])) <= 2) {
251             bobotmapel = 3;
252         } else if
253     (Math.abs(Double.parseDouble(bobotReal[j][3]) -
254     Double.parseDouble(bobotReal[k][3])) <= 4) {
255             bobotmapel = 5;
256         } else if
257     (Math.abs(Double.parseDouble(bobotReal[j][3]) -
258     Double.parseDouble(bobotReal[k][3])) <= 6) {
259             bobotmapel = 7;
260         } else if
261     (Math.abs(Double.parseDouble(bobotReal[j][3]) -
262     Double.parseDouble(bobotReal[k][3])) <= 10) {
263             bobotmapel = 9;
264         }
265     }
266     if
267     (Double.parseDouble(bobotReal[j][3]) <
268     Double.parseDouble(bobotReal[k][3])) {
269         bobotmapel = 1 / bobotmapel;
270     }
271     matrikIng[j][k] = bobotmapel;
272     //tuliskan.write(matrikIng[j][k]+" ");
273     totalbobot[3][k] += bobotmapel;
274 }
275 }
276 //ratabobot[j][3] = totalbobot[3][j] /
277 jumlahJurusan;
278 //tuliskan.newLine();
279 }
280 }
281
282     for (int j = 0; j < jumlahJurusan; j++) {
283         for (int k = 0; k < jumlahJurusan; k++)
284     {
285             matrikIng[j][k] = matrikIng[j][k] /
286     totalbobot[3][k];
287             tuliskan.write(matrikIng[j][k] + " ");
288             totalbobotnorm[3][j] +=
289     matrikIng[j][k];
290         }
291         tuliskan.newLine();
292     }
293
294     for (int j = 0; j < jumlahJurusan; j++) {
295         ratabobot[j][3] = totalbobotnorm[3][j]
296     / jumlahJurusan;
297     }
298
299     tuliskan.newLine();

```

```

300
301 //bobot Mat
302 tulis.write("Matematika");
303 tulis.newLine();
304 for (int j = 0; j < jumlahJurusan; j++) {
305     for (int k = 0; k < jumlahJurusan; k++)
306     {
307         double bobotmapel = 0;
308         if
309 (Math.abs(Double.parseDouble(bobotReal[j][4]) -
310 Double.parseDouble(bobotReal[k][4])) == 0) {
311             bobotmapel = 1;
312         } else if
313 (Math.abs(Double.parseDouble(bobotReal[j][4]) -
314 Double.parseDouble(bobotReal[k][4])) <= 2) {
315             bobotmapel = 3;
316         } else if
317 (Math.abs(Double.parseDouble(bobotReal[j][4]) -
318 Double.parseDouble(bobotReal[k][4])) <= 4) {
319             bobotmapel = 5;
320         } else if
321 (Math.abs(Double.parseDouble(bobotReal[j][4]) -
322 Double.parseDouble(bobotReal[k][4])) <= 6) {
323             bobotmapel = 7;
324         } else if
325 (Math.abs(Double.parseDouble(bobotReal[j][4]) -
326 Double.parseDouble(bobotReal[k][4])) <= 10) {
327             bobotmapel = 9;
328         }
329         if
330 (Double.parseDouble(bobotReal[j][4]) <
331 Double.parseDouble(bobotReal[k][4])) {
332             bobotmapel = 1 / bobotmapel;
333         }
334         matrikMat[j][k] = bobotmapel;
335         //tulis.write(matrikMat[j][k]+" ");
336         totalbobot[4][k] += bobotmapel;
337     }
338     //ratabobot[j][4] = totalbobot[4][j] /
339     jumlahJurusan;
340     //tulis.newLine();
341 }
342
343 for (int j = 0; j < jumlahJurusan; j++) {
344     for (int k = 0; k < jumlahJurusan; k++)
345     {
346         matrikMat[j][k] = matrikMat[j][k] /
347         totalbobot[4][k];
348         tulis.write(matrikMat[j][k] + " ");
349         totalbobotnorm[4][j] +=
350         matrikMat[j][k];
351     }
352     tulis.newLine();
353 }
354
355 for (int j = 0; j < jumlahJurusan; j++) {

```

```

359         ratabobot[j][4] = totalbobotnorm[4][j]
360 / jumlahJurusan;
361     }
362     tulis.newLine();;
363
364     //bobot Fis
365     tulis.write("Fisika");
366     tulis.newLine();
367     for (int j = 0; j < jumlahJurusan; j++) {
368         for (int k = 0; k < jumlahJurusan; k++)
369     {
370             double bobotmapel = 0;
371             if
372 (Math.abs(Double.parseDouble(bobotReal[j][5]) -
373 Double.parseDouble(bobotReal[k][5])) == 0) {
374                 bobotmapel = 1;
375             } else if
376 (Math.abs(Double.parseDouble(bobotReal[j][5]) -
377 Double.parseDouble(bobotReal[k][5])) <= 2) {
378                 bobotmapel = 3;
379             } else if
380 (Math.abs(Double.parseDouble(bobotReal[j][5]) -
381 Double.parseDouble(bobotReal[k][5])) <= 4) {
382                 bobotmapel = 5;
383             } else if
384 (Math.abs(Double.parseDouble(bobotReal[j][5]) -
385 Double.parseDouble(bobotReal[k][5])) <= 6) {
386                 bobotmapel = 7;
387             } else if
388 (Math.abs(Double.parseDouble(bobotReal[j][5]) -
389 Double.parseDouble(bobotReal[k][5])) <= 10) {
390                 bobotmapel = 9;
391             }
392             if
393 (Double.parseDouble(bobotReal[j][5]) <
394 Double.parseDouble(bobotReal[k][5])) {
395                 bobotmapel = 1 / bobotmapel;
396             }
397             matrikFis[j][k] = bobotmapel;
400             // tulis.write(matrikFis[j][k]+"
401 ");
402             totalbobot[5][k] += bobotmapel;
403         }
404
405         //ratabobot[j][5] = totalbobot[5][j] /
406 jumlahJurusan;
407         //tulis.newLine();;
408     }
409
410     for (int j = 0; j < jumlahJurusan; j++) {
411         for (int k = 0; k < jumlahJurusan; k++)
412     {
413             matrikFis[j][k] = matrikFis[j][k] /
414 totalbobot[5][k];
415             tulis.write(matrikFis[j][k] + " ");
416             totalbobotnorm[5][j] +=
417 matrikFis[j][k];

```

```

418     }
419     tulis.newLine();
420 }
421
422     for (int j = 0; j < jumlahJurusan; j++) {
423         ratabobot[j][5] = totalbobotnorm[5][j]
424 / jumlahJurusan;
425     }
426     tulis.newLine();
427
428     //bobot kim
429     tulis.write("Kimia");
430     tulis.newLine();
431     for (int j = 0; j < jumlahJurusan; j++) {
432         for (int k = 0; k < jumlahJurusan; k++)
433 {
434             double bobotmapel = 0;
435             if
436 (Math.abs(Double.parseDouble(bobotReal[j][6])
437 Double.parseDouble(bobotReal[k][6])) == 0) {
438                 bobotmapel = 1;
439             } else if
440 (Math.abs(Double.parseDouble(bobotReal[j][6])
441 Double.parseDouble(bobotReal[k][6])) <= 2) {
442                 bobotmapel = 3;
443             } else if
444 (Math.abs(Double.parseDouble(bobotReal[j][6])
445 Double.parseDouble(bobotReal[k][6])) <= 4) {
446                 bobotmapel = 5;
447             } else if
448 (Math.abs(Double.parseDouble(bobotReal[j][6])
449 Double.parseDouble(bobotReal[k][6])) <= 6) {
450                 bobotmapel = 7;
451             } else if
452 (Math.abs(Double.parseDouble(bobotReal[j][6])
453 Double.parseDouble(bobotReal[k][6])) <= 10) {
454                 bobotmapel = 9;
455             }
456             if
457 (Double.parseDouble(bobotReal[j][6])
458 Double.parseDouble(bobotReal[k][6])) <
459 Double.parseDouble(bobotReal[k][6])) {
460                 bobotmapel = 1 / bobotmapel;
461             }
462
463             matrikKim[j][k] = bobotmapel;
464             // tulis.write(matrikKim[j][k]+"
465 ");
466             totalbobot[6][k] += bobotmapel;
467         }
468
469         //ratabobot[j][6] = totalbobot[6][j] /
470 jumlahJurusan;
471         //tulis.newLine();
472     }
473
474     for (int j = 0; j < jumlahJurusan; j++) {
475         for (int k = 0; k < jumlahJurusan; k++)
476 {

```

```

477         matrikKim[j][k] = matrikKim[j][k] /
478 totalbobot[6][k];
479         tulis.write(matrikKim[j][k] + " ");
480         totalbobotnorm[6][j] +=
481 matrikKim[j][k];
482     }
483     tulis.newLine();;
484 }
485
486     for (int j = 0; j < jumlahJurusan; j++) {
487         ratabobot[j][6] = totalbobotnorm[6][j]
488 / jumlahJurusan;
489     }
490     tulis.newLine();;
491
492     //bobot Bio
493     tulis.write("Bio");
494     tulis.newLine();
495     for (int j = 0; j < jumlahJurusan; j++) {
496         for (int k = 0; k < jumlahJurusan; k++)
497     {
498             double bobotmapel = 0;
499             if
500 (Math.abs(Double.parseDouble(bobotReal[j][7]) -
501 Double.parseDouble(bobotReal[k][7])) == 0) {
502                 bobotmapel = 1;
503             } else if
504 (Math.abs(Double.parseDouble(bobotReal[j][7]) -
505 Double.parseDouble(bobotReal[k][7])) <= 2) {
506                 bobotmapel = 3;
507             } else if
508 (Math.abs(Double.parseDouble(bobotReal[j][7]) -
509 Double.parseDouble(bobotReal[k][7])) <= 4) {
510                 bobotmapel = 5;
511             } else if
512 (Math.abs(Double.parseDouble(bobotReal[j][7]) -
513 Double.parseDouble(bobotReal[k][7])) <= 6) {
514                 bobotmapel = 7;
515             } else if
516 (Math.abs(Double.parseDouble(bobotReal[j][7]) -
517 Double.parseDouble(bobotReal[k][7])) <= 10) {
518                 bobotmapel = 9;
519             }
520
521             if
522 (Double.parseDouble(bobotReal[j][7]) <
523 Double.parseDouble(bobotReal[k][7])) {
524                 bobotmapel = 1 / bobotmapel;
525             }
526
527             matrikBio[j][k] = bobotmapel;
528             // tulis.write(matrikBio[j][k]+
529 ");
530             totalbobot[7][k] += bobotmapel;
531         }
532     }
533     //tulis.newLine();;
534 }
535     for (int j = 0; j < jumlahJurusan; j++) {

```

```

536         for (int k = 0; k < jumlahJurusan; k++)
537     {
538         matrikBio[j][k] = matrikBio[j][k] /
539     totalbobot[7][k];
540         tulis.write(matrikBio[j][k] + " ");
541         totalbobotnorm[7][j] +=
542     matrikBio[j][k];
543     }
544     tulis.newLine();
545     }
546     for (int j = 0; j < jumlahJurusan; j++) {
547         ratabobot[j][7] = totalbobotnorm[7][j]
548     / jumlahJurusan;
549     }
550
551     tulis.newLine();
552     //bobot Sejarah
553     tulis.write("Sejarah");
554     tulis.newLine();
555     for (int j = 0; j < jumlahJurusan; j++) {
556         for (int k = 0; k < jumlahJurusan; k++)
557     {
558             double bobotmapel = 0;
559             if
560     (Math.abs(Double.parseDouble(bobotReal[j][8]) -
561     Double.parseDouble(bobotReal[k][8])) == 0) {
562                 bobotmapel = 1;
563             } else if
564     (Math.abs(Double.parseDouble(bobotReal[j][8]) -
565     Double.parseDouble(bobotReal[k][8])) <= 2) {
566                 bobotmapel = 3;
567             } else if
568     (Math.abs(Double.parseDouble(bobotReal[j][8]) -
569     Double.parseDouble(bobotReal[k][8])) <= 4) {
570                 bobotmapel = 5;
571             } else if
572     (Math.abs(Double.parseDouble(bobotReal[j][8]) -
573     Double.parseDouble(bobotReal[k][8])) <= 6) {
574                 bobotmapel = 7;
575             } else if
576     (Math.abs(Double.parseDouble(bobotReal[j][8]) -
577     Double.parseDouble(bobotReal[k][8])) <= 10) {
578                 bobotmapel = 9;
579             }
580
581             if
582     (Double.parseDouble(bobotReal[j][8]) <
583     Double.parseDouble(bobotReal[k][8])) {
584                 bobotmapel = 1 / bobotmapel;
585             }
586
587             matrikSjr[j][k] = bobotmapel;
588             //tulis.write(matrikSjr[j][k]+" ");
589             totalbobot[8][k] += bobotmapel;
590         }
591     }
592     //tulis.newLine();
593 }
594

```

```

595     for (int j = 0; j < jumlahJurusan; j++) {
596         for (int k = 0; k < jumlahJurusan; k++)
597     {
598         matrikSjr[j][k] = matrikSjr[j][k] /
599 totalbobot[8][k];
600         tulis.write(matrikSjr[j][k] + " ");
601         totalbobotnorm[8][j] +=
602 matrikSjr[j][k];
603     }
604     tulis.newLine();
605 }
606
607     for (int j = 0; j < jumlahJurusan; j++) {
608         ratabobot[j][8] = totalbobotnorm[8][j]
609 / jumlahJurusan;
610     }
611     tulis.newLine();
612
613     //bobot seni budaya
614     tulis.write("Seni Budaya");
615     tulis.newLine();
616     for (int j = 0; j < jumlahJurusan; j++) {
617         for (int k = 0; k < jumlahJurusan; k++)
618     {
619             double bobotmapel = 0;
620             if
621 (Math.abs(Double.parseDouble(bobotReal[j][9])
622 Double.parseDouble(bobotReal[k][9])) == 0) {
623                 bobotmapel = 1;
624             } else if
625 (Math.abs(Double.parseDouble(bobotReal[j][9])
626 Double.parseDouble(bobotReal[k][9])) <= 2) {
627                 bobotmapel = 3;
628             } else if
629 (Math.abs(Double.parseDouble(bobotReal[j][9])
630 Double.parseDouble(bobotReal[k][9])) <= 4) {
631                 bobotmapel = 5;
632             } else if
633 (Math.abs(Double.parseDouble(bobotReal[j][9])
634 Double.parseDouble(bobotReal[k][9])) <= 6) {
635                 bobotmapel = 7;
636             } else if
637 (Math.abs(Double.parseDouble(bobotReal[j][9])
638 Double.parseDouble(bobotReal[k][9])) <= 10) {
639                 bobotmapel = 9;
640             }
641
642             if
643 (Double.parseDouble(bobotReal[j][9])
644 Double.parseDouble(bobotReal[k][9])) <
645                 bobotmapel = 1 / bobotmapel;
646             }
647
648             matrikSnb[j][k] = bobotmapel;
649             //tulis.write(matrikSnb[j][k]+" ");
650             totalbobot[9][k] += bobotmapel;
651         }
652
653         //tulis.newLine();

```



```

654     }
655     for (int j = 0; j < jumlahJurusan; j++) {
656         for (int k = 0; k < jumlahJurusan; k++)
657     {
658         matrikSnb[j][k] = matrikSnb[j][k] /
659 totalbobot[9][k];
660         tulis.write(matrikSnb[j][k] + " ");
661         totalbobotnorm[9][j] +=
662 matrikSnb[j][k];
663     }
664     tulis.newLine();
665 }
666
667     for (int j = 0; j < jumlahJurusan; j++) {
668         ratabobot[j][9] = totalbobotnorm[9][j]
669 / jumlahJurusan;
670     }
671     tulis.newLine();
672
673     //bobot penjaskes
674     tulis.write("Penjaskes");
675     tulis.newLine();
676     for (int j = 0; j < jumlahJurusan; j++) {
677         for (int k = 0; k < jumlahJurusan; k++)
678     {
679         double bobotmapel = 0;
680         if
681 (Math.abs(Double.parseDouble(bobotReal[j][10])
682 Double.parseDouble(bobotReal[k][10])) == 0) {
683             bobotmapel = 1;
684         } else if
685 (Math.abs(Double.parseDouble(bobotReal[j][10])
686 Double.parseDouble(bobotReal[k][10])) <= 2) {
687             bobotmapel = 3;
688         } else if
689 (Math.abs(Double.parseDouble(bobotReal[j][10])
690 Double.parseDouble(bobotReal[k][10])) <= 4) {
691             bobotmapel = 5;
692         } else if
693 (Math.abs(Double.parseDouble(bobotReal[j][10])
694 Double.parseDouble(bobotReal[k][10])) <= 6) {
695             bobotmapel = 7;
696         } else if
697 (Math.abs(Double.parseDouble(bobotReal[j][10])
698 Double.parseDouble(bobotReal[k][10])) <= 10) {
699             bobotmapel = 9;
700         }
701     }
702     if
703 (Double.parseDouble(bobotReal[j][10]) <
704 Double.parseDouble(bobotReal[k][10])) {
705         bobotmapel = 1 / bobotmapel;
706     }
707     matrikPenjas[j][k] = bobotmapel;
708     //tulis.write(matrikPenjas[j][k]+"
709 ");
710     totalbobot[10][j] += bobotmapel;
711
712

```

```

713     }
714
715     //tuliskan newline();
716     }
717
718     for (int j = 0; j < jumlahJurusan; j++) {
719         for (int k = 0; k < jumlahJurusan; k++)
720     {
721             matrikPenjas[j][k] =
722 matrikPenjas[j][k] / totalbobot[10][k];
723             tulis.write(matrikPenjas[j][k] + "
724 ");
725             totalbobotnorm[10][k] +=
726 matrikPenjas[j][k];
727         }
728         tulis.newLine();
729     }
730
731     for (int j = 0; j < jumlahJurusan; j++) {
732         ratabobot[j][10] =
733 totalbobotnorm[10][j] / jumlahJurusan;
734     }
735     tulis.newLine();
736
737     //bobot tik
738     tulis.write("TIK");
739     tulis.newLine();
740     for (int j = 0; j < jumlahJurusan; j++) {
741         for (int k = 0; k < jumlahJurusan; k++)
742     {
743             double bobotmapel = 0;
744             if
745 (Math.abs(Double.parseDouble(bobotReal[j][11]) -
746 Double.parseDouble(bobotReal[k][11])) == 0) {
747                 bobotmapel = 1;
748             } else if
749 (Math.abs(Double.parseDouble(bobotReal[j][11]) -
750 Double.parseDouble(bobotReal[k][11])) <= 2) {
751                 bobotmapel = 3;
752             } else if
753 (Math.abs(Double.parseDouble(bobotReal[j][11]) -
754 Double.parseDouble(bobotReal[k][11])) <= 4) {
755                 bobotmapel = 5;
756             } else if
757 (Math.abs(Double.parseDouble(bobotReal[j][11]) -
758 Double.parseDouble(bobotReal[k][11])) <= 6) {
759                 bobotmapel = 7;
760             } else if
761 (Math.abs(Double.parseDouble(bobotReal[j][11]) -
762 Double.parseDouble(bobotReal[k][11])) <= 10) {
763                 bobotmapel = 9;
764             }
765             if
766 (Double.parseDouble(bobotReal[j][11]) <
767 Double.parseDouble(bobotReal[k][11])) {
768                 bobotmapel = 1 / bobotmapel;
769             }
770         }
771     }

```

```

772
773         matrikTIK[j][k] = bobotmapel;
774         //tuliskan.write(matrikTIK[j][k]+" ");
775         totalbobot[11][k] += bobotmapel;
776     }
777
778     //tuliskan.newLine();;
779 }
780
781 for (int j = 0; j < jumlahJurusan; j++) {
782     for (int k = 0; k < jumlahJurusan; k++)
783 {
784         matrikTIK[j][k] = matrikTIK[j][k] /
785 totalbobot[11][k];
786         tuliskan.write(matrikTIK[j][k] + " ");
787         totalbobotnorm[11][j] +=
788 matrikTIK[j][k];
789     }
790     tuliskan.newLine();;
791 }
792
793     for (int j = 0; j < jumlahJurusan; j++) {
794         ratabobot[j][11] =
795 totalbobotnorm[11][j] / jumlahJurusan;
796     }
797     tuliskan.newLine();;
798
799     //bobot kba
800     tuliskan.write("KBA");
801     tuliskan.newLine();
802     for (int j = 0; j < jumlahJurusan; j++) {
803         for (int k = 0; k < jumlahJurusan; k++)
804 {
805             double bobotmapel = 0;
806             if
807 (Math.abs(Double.parseDouble(bobotReal[j][12]) -
808 Double.parseDouble(bobotReal[k][12])) == 0) {
809                 bobotmapel = 1;
810             } else if
811 (Math.abs(Double.parseDouble(bobotReal[j][12]) -
812 Double.parseDouble(bobotReal[k][12])) <= 2) {
813                 bobotmapel = 3;
814             } else if
815 (Math.abs(Double.parseDouble(bobotReal[j][12]) -
816 Double.parseDouble(bobotReal[k][12])) <= 4) {
817                 bobotmapel = 5;
818             } else if
819 (Math.abs(Double.parseDouble(bobotReal[j][12]) -
820 Double.parseDouble(bobotReal[k][12])) <= 6) {
821                 bobotmapel = 7;
822             } else if
823 (Math.abs(Double.parseDouble(bobotReal[j][12]) -
824 Double.parseDouble(bobotReal[k][12])) <= 10) {
825                 bobotmapel = 9;
826             }
827
828             if
829 (Double.parseDouble(bobotReal[j][12]) <
830 Double.parseDouble(bobotReal[k][12])) {

```

```

831         bobotmapel = 1 / bobotmapel;
832     }
833
834     matrikKba[j][k] = bobotmapel;
835     //tuliskan.write(matrikKba[j][k] + "
836 ");
837     totalbobot[12][k] += bobotmapel;
838 }
839 }
840
841 for (int j = 0; j < jumlahJurusan; j++) {
842     for (int k = 0; k < jumlahJurusan; k++)
843     {
844         matrikKba[j][k] = matrikKba[j][k] /
845 totalbobot[12][k];
846         tuliskan.write(matrikKba[j][k] + " ");
847         totalbobotnorm[12][j] +=
848 matrikKba[j][k];
849     }
850     tuliskan.newLine();
851 }
852 for (int j = 0; j < jumlahJurusan; j++) {
853     ratabobot[j][12] =
854 totalbobotnorm[12][j] / jumlahJurusan;
855 }
856     tuliskan.newLine();

```

**Gambar 5.8 Pembuatan Matriks Mata Pelajaran Model 1**

Penjelasan untuk Gambar 5.8 adalah sebagai berikut :

Langkah awal adalah inisialisasi variabel matriks mata pelajaran yang akan menampung matriks perbandingan berpasangan untuk tiap mata pelajaran. Kemudian dilakukan proses pengisian matriks dengan membandingkan nilai bobot mata pelajaran yang telah didapatkan sebelumnya. Pembuatan matriks perbandingan berpasangan dilakukan per mata pelajaran dengan membandingkan bobot nilai mata pelajaran tersebut untuk tiap fakultas. Untuk model 1 di atas dilakukan pembuatan matriks perbandingan berpasangan untuk mata pelajaran agama, kwn, Bahasa Indonesia, Bahasa Inggris, matematika, fisika, kimia, biologi, sejarah, seni budaya, penjasKes, TIK, dan kemampuan Bahasa asing. Di dalam proses pembuatan matriks perbandingan berpasangan juga dilakukan proses pengubahan matriks tersebut menjadi matriks normalisasi. Setelah diubah ke dalam bentuk normalisasi, dihitung rata-rata bobot mata pelajaran untuk tiap jurusan yang akan digunakan untuk perhitungan persentase akhir.

#### f. Menghitung Persentase Akhir Untuk Tiap Jurusan

Di dalam program terdapat source code untuk menghitung persentase akhir untuk tiap jurusan dalam fakultas tertentu. Implementasi dapat dilihat pada Gambar 5.9

```

1 //tampilan matriks final
2 tuliskan.write("final");
3 tuliskan.newLine();
4 for (int j = 0; j < ratabobot.length; j++) {
5     for (int k = 0; k < ratabobot[0].length; k++) {

```

```

6         tulis.write(ratabobot[j][k] + " ");
7     }
8     tulis.newLine();;
9 }
10
11 double totaljurusan[] = new double[jumlahJurusan];
12
13 //matriks akhir
14 tulis.write("final banget");
15 tulis.newLine();
16 double finalfix[][] = new double[jumlahJurusan][13];
17
18 for (int j = 0; j < finalfix.length; j++) {
19     for (int k = 0; k < finalfix[0].length; k++) {
20         finalfix[j][k] = rataabobot[j][k] * rata[k];
21         tulis.write(finalfix[j][k] + " ");
22         totaljurusan[j] += finalfix[j][k];
23     }
24     tulis.newLine();;
25 }
26
27 tulis.write("\nHasil\n");
28 tulis.newLine();
29
30 int max = 0;
31
32 for (int j = 0; j < jumlahJurusan; j++) {
33     tulis.write((j + 1) + " : " + totaljurusan[j]);
34 }
35
36 for (int k = 0; k < totaljurusan.length; k++) {
37     if (totaljurusan[max] < totaljurusan[k]) {
38         max = k;
39     }
40 }
41
42 tulis.newLine();
43 String jurusan = "";
44
45 tulis.write("Hasil Jurusan : ");
46 tulis.newLine();
47
48 jurusan = daftarJurusanUrut[max];
49
50 tulis.write(jurusan);
51 tulisHasil.write(jurusan);
52 tulisHasil.newLine();
53
54 tulis.newLine();
55
56 tulis.write("=====");
57 tulis.newLine();
58 tulis.newLine();

```

**Gambar 5.9 Perhitungan Persentase Akhir Jurusan**

Penjelasan untuk Gambar 5.9 adalah sebagai berikut :

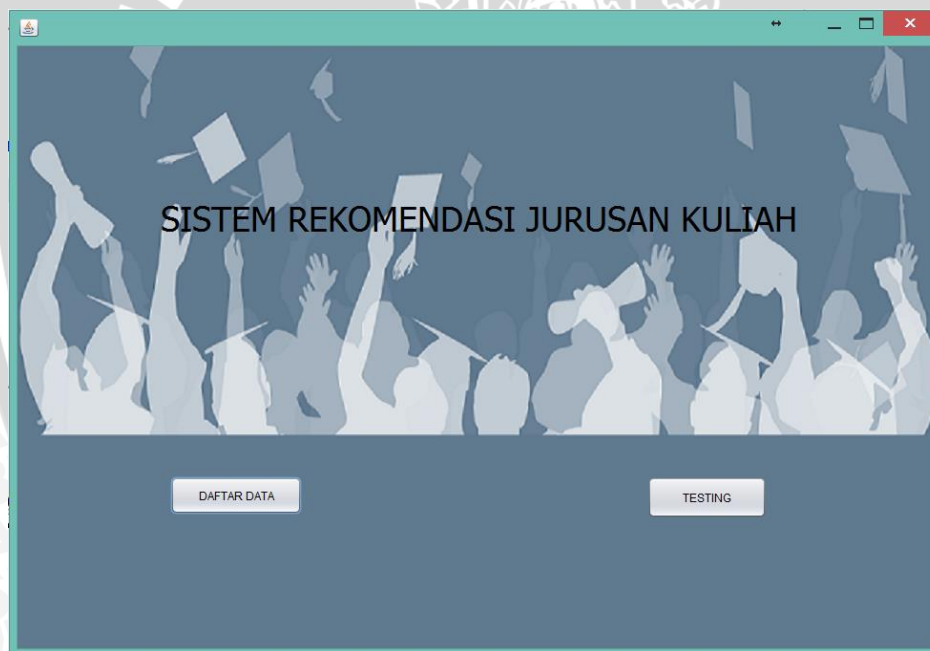
Langkah awal adalah dengan membuat array finalfix untuk menampung proses perhitungan persentase akhir. Proses perhitungan persentase akhir dilakukan dengan mengalikan rata-rata bobot mata pelajaran pada nilai data uji dengan rata-rata bobot mata pelajaran untuk tiap jurusan. Setelah didapat hasil perkalian, maka hasil perkalian tiap mata pelajaran tiap jurusan dijumlahkan untuk mendapatkan persentase akhir. Jurusan dengan persentase akhir tertinggi akan dipilih untuk menjadi hasil rekomendasi.

### 5.3 Implementasi *User Interface*

Pada implementasi *user interface* akan dibuat 7 halaman yaitu halaman utama, halaman daftar data, halaman *setting* variabel, halaman *testing Naïve Bayes*, halaman manualisasi *Naïve Bayes*, halaman *testing AHP*, dan halaman manualisasi *AHP*.

#### 5.3.1 Implementasi Halaman Utama

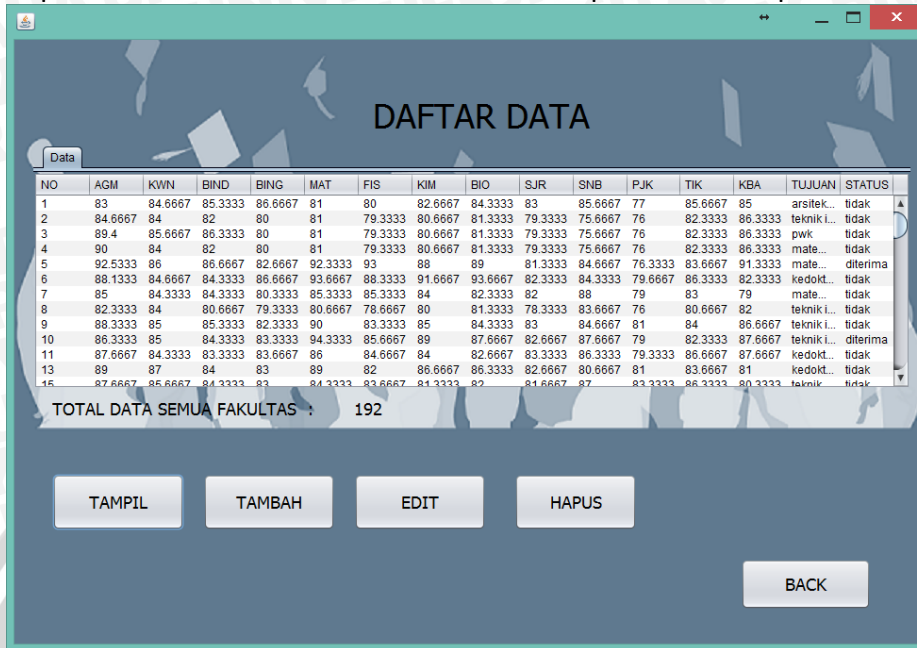
Implementasi halaman utama dapat dilihat pada gambar 5.10



Gambar 5.10 Implementasi Halaman Utama

### 5.3.2 Implementasi Halaman Daftar Data

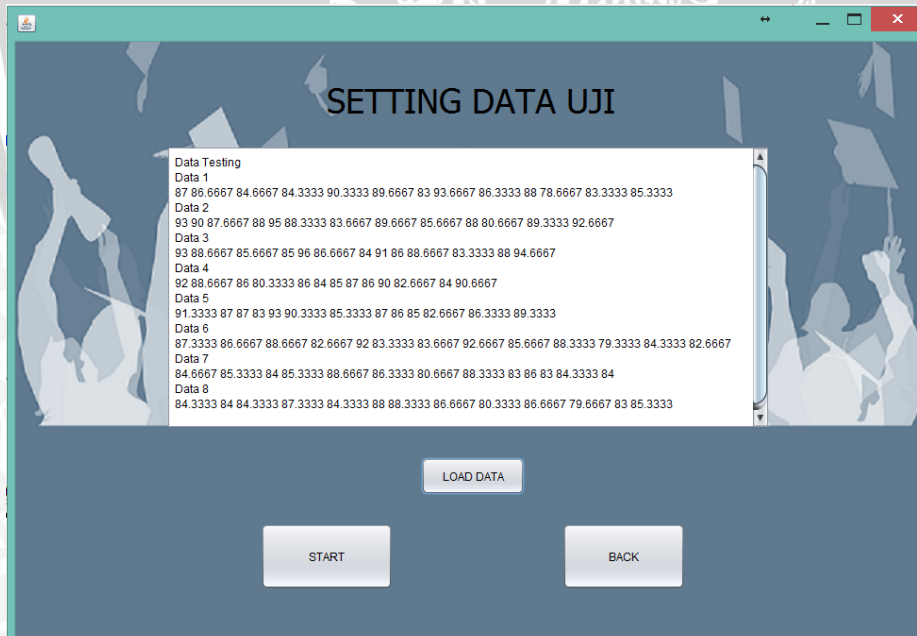
Implementasi halaman daftar data dapat dilihat pada Gambar 5.11



Gambar 5.11 Implementasi Halaman Daftar Data

### 5.3.3 Implementasi Halaman Setting Variabel

Implementasi halaman setting variabel dapat dilihat pada Gambar 5.12



Gambar 5.12 Implementasi Halaman Setting Variabel

### 5.3.4 Implementasi Halaman *Testing Naïve Bayes*

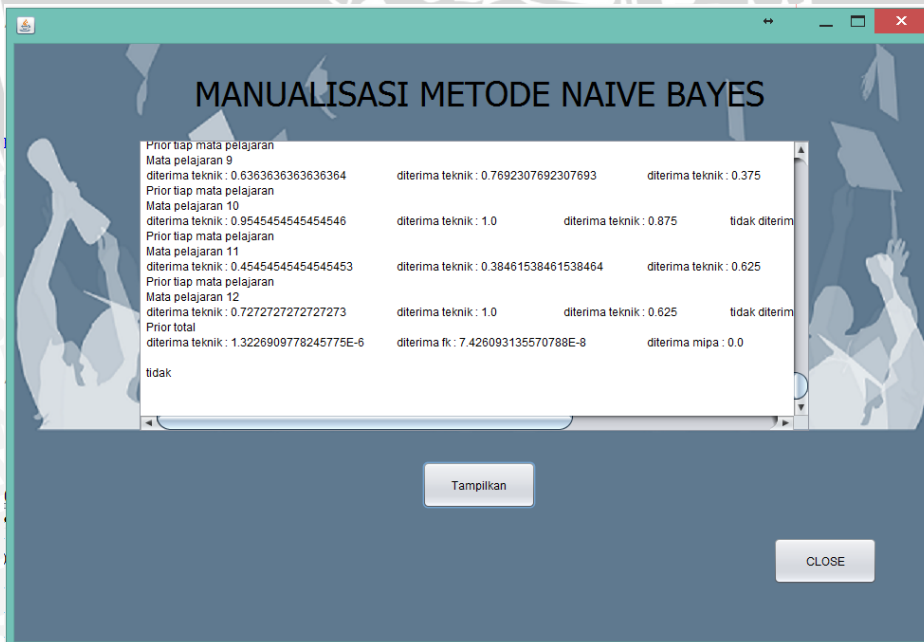
Implementasi halaman *testing baive bayes* dapat dilihat pada gambar 5.13



Gambar 5.13 Implementasi Halaman Testing Naïve Bayes

### 5.3.5 Implementasi Halaman *Manualisasi Naïve Bayes*

Implementasi halaman manualisasi *naïve bayes* dapat dilihat pada Gambar 5.14

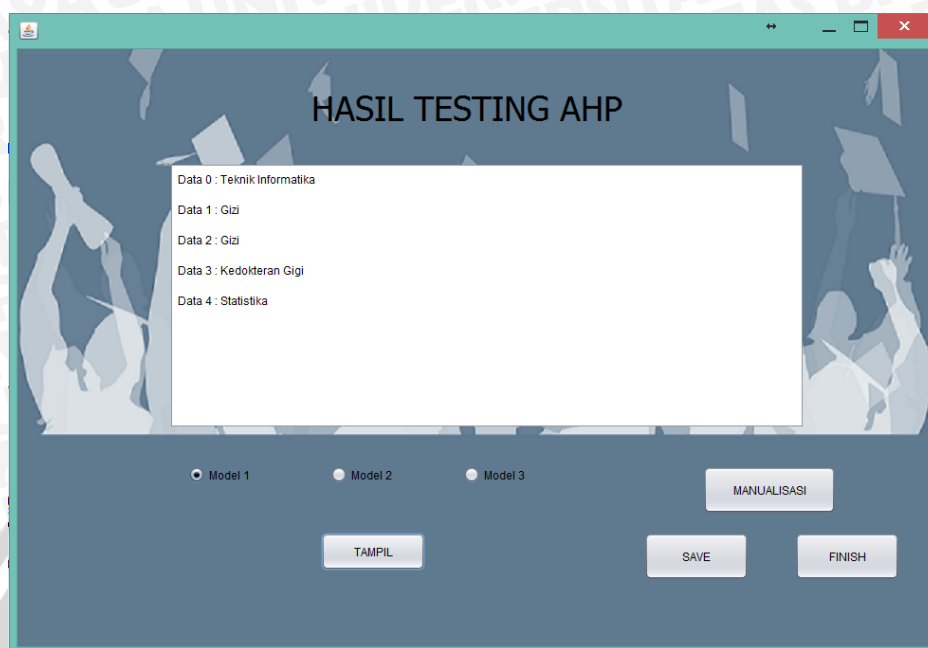


Gambar 5.14 Implementasi Halaman Manualisasi Naïve Bayes



### 5.3.6 Implementasi Halaman *Testing AHP*

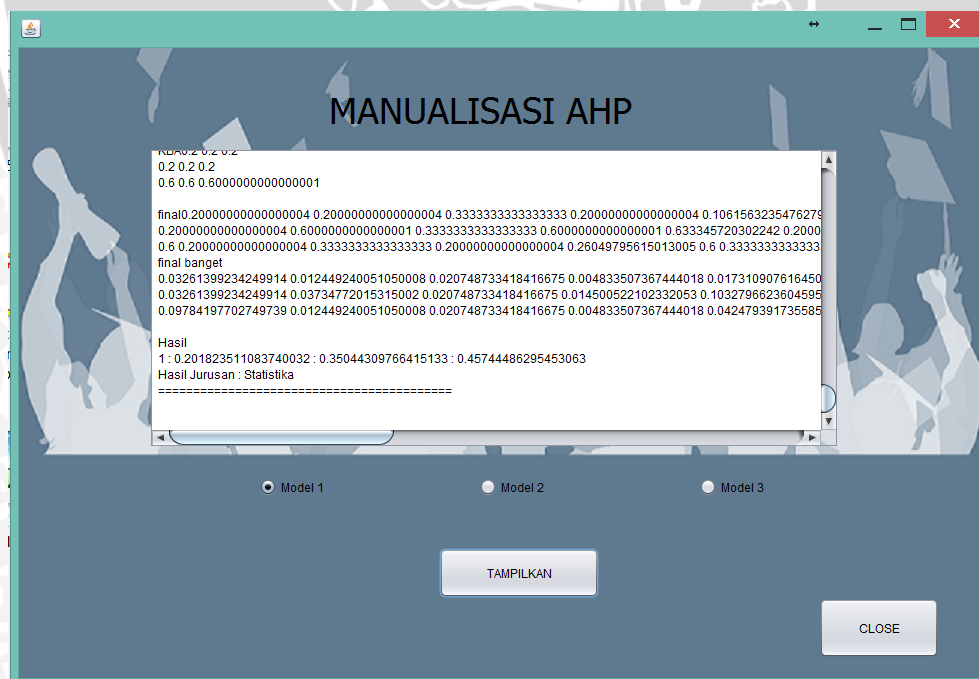
Implementasi halaman *testing AHP* dapat dilihat pada Gambar 5.15



Gambar 5.15 Implementasi Halaman *Testing AHP*

### 5.3.7 Implementasi Halaman *Manualisasi AHP*

Implementasi halaman *manualisasi AHP* dapat dilihat pada gambar 5.16



Gambar 5.16 Implementasi Halaman *Manualisasi AHP*

## BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini akan membahas mengenai pengujian serta analisis terhadap implementasi sistem pengklasifikasi jurusan kuliah untuk anak SMA menggunakan algoritma *AHP-Naïve Bayes*.

### 6.1 Hasil dan Analisa Jumlah Data Latih Algoritma *Naïve Bayes*

Pada uji coba jumlah data latih untuk algoritma *naïve bayes* digunakan data pelajar SMAN 1 Malang. Untuk tiap percobaan, digunakan jumlah data latih dengan persentase yang berbeda untuk mengetahui jumlah persentase data latih dengan nilai keakuratan tertinggi. Data uji berasal dari data latih yang tidak digunakan sebagai data latih di dalam uji coba. Untuk data uji, digunakan data dengan kelas tidak dikarenakan data dengan kelas tidak merupakan data dengan jumlah tertinggi. Hal ini dilakukan agar jumlah data latih untuk kelas lainnya tidak berkurang secara signifikan dikarenakan keterbatasan data. Total data keseluruhan adalah 192 data, dengan batasan pengujian adalah pengujian yang dilakukan dengan 1 variasi saja, yaitu pengujian berdasarkan skenario hanya dilakukan sebanyak 1 kali. Untuk hasil uji coba jumlah data latih algoritma *naïve bayes* dapat dilihat pada Tabel 6.1

**Tabel 6.1 Hasil Uji Coba Jumlah Data Latih Algoritma *Naïve Bayes***

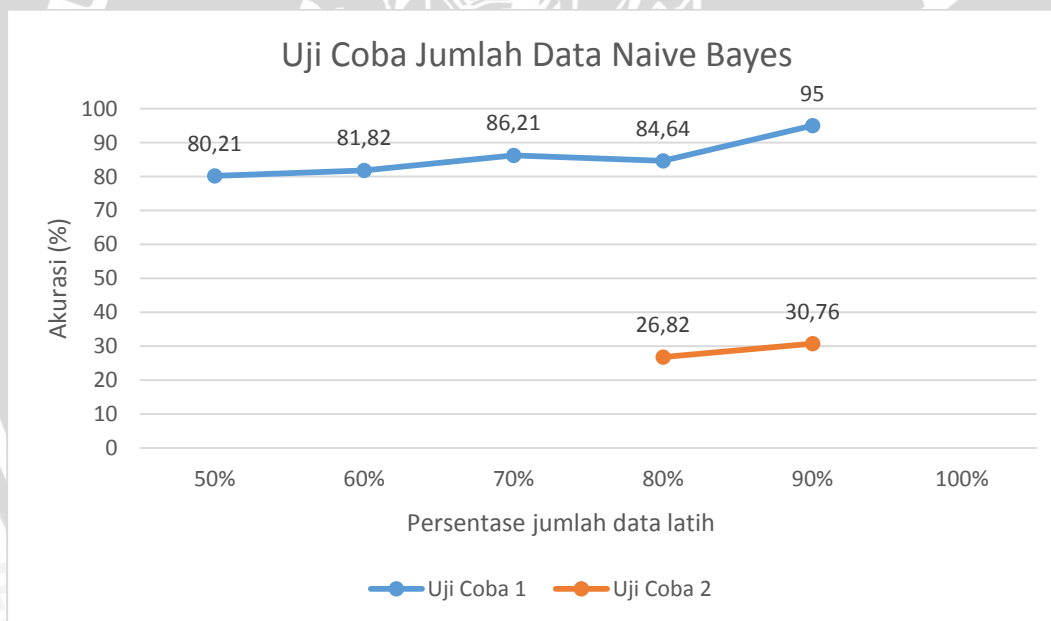
Jumlah Data Latih (persen)	Jumlah Data Uji (persen)	Hasil		Persentase
		Jumlah Benar	Jumlah Salah	
50	50	77	19	80,21 %
60	40	63	14	81,82 %
70	30	50	8	86,21 %
80	20	33	6	84,64 %
90	10	19	1	95 %

Pada tabel di atas dapat dilihat bahwa akurasi terbaik didapat ketika jumlah data latih yang digunakan sebanyak 90% dari total data, dan jumlah data uji sebanyak 10%. Jika dilihat dari hasil akurasi, dapat dikatakan semakin banyak data latih yang dilibatkan, semakin tinggi juga akurasi yang bisa didapatkan dari sistem. Namun perlu diperhatikan, bahwa akurasi 95% didapat dikarenakan jumlah data latih tidak lebih banyak dibandingkan dengan data kelas lain, sehingga jika data uji berasal dari kelas diterima fakultas, akurasi yang didapat belum tentu sama. Berikut adalah hasil dari pengujian dengan data uji merupakan data dari kelas diterima di masing-masing fakultas yang dapat dilihat pada Tabel 6.2. Untuk pengujian ini hanya terdapat 2 skenario dikarenakan keterbatasan jumlah data dari kelas diterima di masing-masing fakultas.

**Tabel 6.2 Hasil Uji Coba Jumlah Data Latih Algoritma Naïve Bayes Skenario 2**

Jumlah Data Latih (persen)	Jumlah Data Uji (persen)	Hasil		Persentase
		jumlah benar	jumlah salah	
80	20	11	30	26,82 %
90	10	8	18	30,76 %

Dari tabel di atas menunjukkan, bahwa ketika data uji adalah campuran data kelas diterima tiap fakultas dengan data kelas tidak, maka akurasi akan berubah sangat drastis dengan akurasi maksimum 30,76%. Hal ini menunjukkan bahwa metode naïve bayes sangat bergantung pada kualitas data latih, semakin seimbang jumlah data tiap kelas, maka semakin baik juga akurasi sistem. Uji coba kedua juga menunjukkan, bahwa semakin banyak data latih yang digunakan, maka semakin baik juga akurasi sistem. Untuk perbandingan grafik yang dihasilkan dari uji coba ini dapat dilihat pada Gambar 6.1



**Gambar 6.1 Grafik Hasil Akurasi Uji Coba Jumlah Data**

## 6.2 Hasil dan Analisa Jumlah Data Latih Algoritma Naïve Bayes dengan Data Latih Diskrit

Pada uji coba jumlah data latih algoritma *naïve bayes* dengan data latih diskrit, data nilai akademik yg bertipe data kontinu dirubah ke dalam data diskrit untuk mempermudah perhitungan. Pengujian ini bertujuan untuk mengetahui perbandingan akurasi antara sistem yang menggunakan data bertipe kontinu

dengan sistem yang menggunakan data bertipe diskrit. Untuk mengubah data kontinu ke dalam bentuk diskrit, digunakan range yang dijelaskan pada Tabel 6.3

**Tabel 6.3 Keterangan Range Nilai**

Range	Nilai Diskrit
Nilai < 70	Rendah
70 <= Nilai <= 85	Sedang
Nilai > 85	Tinggi

Untuk data latih pada contoh perhitungan ini, digunakan data sample yang terdapat pada Tabel 3.2 dan untuk data latih digunakan data latih yang telah didefinisikan pada Tabel 3.3. Berdasarkan Tabel 6.3, data diskrit memiliki 3 kategori, yaitu TINGGI (untuk nilai > 85), SEDANG (untuk 70 <= nilai <= 85), dan RENDAH (untuk nilai < 70). Dalam contoh pengubahan, digunakan data di dalam Tabel 3.2. Nilai mata pelajaran matematika pada data pertama memiliki nilai 92.6667, sehingga memiliki nilai diskrit TINGGI karena memiliki nilai lebih dari 80. Untuk hasil lengkap dari pengolahan data menjadi data diskrit dapat dilihat pada Tabel 6.4 :

**Tabel 6.4 Data yang Telah Diolah**

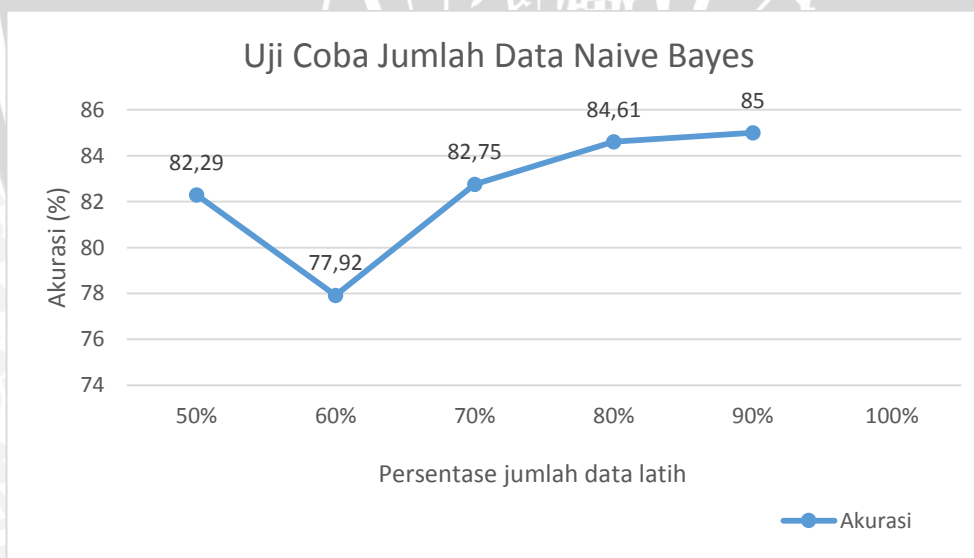
Matematika	Fisika	Kimia	Biologi	TIK	Bind	Bing	Hasil
TINGGI	TINGGI	TINGGI	TINGGI	TINGGI	TINGGI	TINGGI	Diterima
SEDANG	SEDANG	SEDANG	TINGGI	SEDANG	TINGGI	SEDANG	Diterima
TINGGI	SEDANG	SEDANG	TINGGI	TINGGI	TINGGI	TINGGI	Diterima
TINGGI	SEDANG	SEDANG	TINGGI	TINGGI	TINGGI	SEDANG	Diterima
TINGGI	TINGGI	TINGGI	TINGGI	SEDANG	SEDANG	SEDANG	Diterima
TINGGI	TINGGI	SEDANG	TINGGI	SEDANG	SEDANG	SEDANG	Tidak
TINGGI	SEDANG	SEDANG	TINGGI	SEDANG	TINGGI	SEDANG	Tidak
TINGGI	SEDANG	TINGGI	TINGGI	SEDANG	SEDANG	SEDANG	Tidak
TINGGI	SEDANG	SEDANG	TINGGI	TINGGI	TINGGI	SEDANG	Tidak
TINGGI	TINGGI	SEDANG	TINGGI	TINGGI	SEDANG	SEDANG	Tidak

Kemudian setelah dilakukan pengubahan data latih, dilakukan perhitungan untuk menentukan klasifikasi dari data uji dengan alur proses yang telah dijelaskan pada Gambar 3.3. Khusus untuk perhitungan likelihood, untuk data diskrit digunakan persamaan 2.10. Setelah mendefinisikan cara perhitungan kemungkinan dengan data latih diskrit, dilakukan uji coba jumlah data latih algoritma *naïve bayes* dengan batasan pengujian adalah pengujian dilakukan dengan 1 variasi, yaitu perhitungan akurasi dilakukan sekali untuk tiap scenario. Hasil dari pengujian dijabarkan dalam Tabel 6.5

**Tabel 6.5 Hasil Uji Coba Jumlah Data Latih Algoritma Naïve Bayes dengan Data Latih Diskrit**

Jumlah Data Latih (persen)	Jumlah Data Uji	Hasil		Persentase
		Jumlah Benar	Jumlah Salah	
50	50	77	17	82,29 %
60	40	60	17	77,92 %
70	30	48	10	82,75 %
80	20	33	6	84,61 %
90	10	17	3	85 %

Dari hasil pengujian tersebut, dapat dikatakan bahwa sistem dengan data latih bertipe diskrit memiliki akurasi maksimum 85% yang didapat pada keadaan jumlah data latih sebanyak 90% dan data uji sebanyak 10%. Akurasi ini masih berada di bawah akurasi dari sistem dengan data latih bertipe kontinu. Hal ini dapat disebabkan karena range untuk menentukan nilai data diskrit belum tentu range dengan nilai yang optimum, sehingga menimbulkan nilai akurasi yang berada di bawah perhitungan dengan tipe data kontinu. Dengan hasil ini, dapat disimpulkan untuk klasifikasi fakultas dengan data nilai, sistem dengan data latih bertipe kontinu lebih bisa diandalkan dibandingkan sistem dengan data latih bertipe diskrit. Untuk penjelasan akurasi secara grafik dapat dilihat pada Gambar 6.2



**Gambar 6.2 Grafik Hasil Akurasi Uji Coba Jumlah Data dengan Data Latih Diskrit**

### 6.3 Hasil dan Analisa Uji Coba Konsistensi Hasil Rekomendasi AHP

Pada uji coba konsistensi hasil rekomendasi AHP, dilakukan 10 perhitungan menggunakan data latih dan perhitungan dilakukan menggunakan 3 model perhitungan algoritma AHP yang berbeda. Untuk penjelasan model AHP dapat dilihat pada Tabel 3.6. Tingkat konsistensi yang tinggi antar 3 model tersebut dapat membuat *user* untuk mendapatkan rekomendasi yang lebih baik. Daftar data yang diujikan dapat dilihat pada Tabel 6.6

**Tabel 6.6 Daftar Data Uji Coba Konsistensi Hasil Rekomendasi AHP**

Matematika	Fisika	Kimia	Biologi	TIK	Bind	Bing	Hasil
90.6667	88.6667	84.3333	84.3333	91.6667	89	87	teknik sipil
89	89.3333	88.3333	84.6667	96	91.6667	84	arsitektur
90.6667	87.3333	89.3333	81.3333	98.6667	88.6667	83	kedokteran gigi
93	88.6667	85.6667	85	96	86.6667	84	gizi
92	87	86.6667	82.6667	92.3333	92.6667	87.3333	matematika
85	87	86.6667	80	94.3333	86.6667	80	teknik material dan metalurgi
87.3333	86.6667	86.6667	79.6667	92.3333	88.3333	81.6667	teknik mesin
87	86.6667	84.6667	84.3333	90.3333	89.6667	83	Kedokteran
91.6667	86.6667	90	83.3333	92.3333	85	82.3333	teknik industri
92.5333	86	86.6667	82.6667	92.3333	93	88	Matematika

Data tersebut dihitung dengan menggunakan metode AHP untuk tiap model. Pengujian metode AHP dilakukan terpisah dengan metode naïve bayes untuk mengetahui kinerja metode AHP secara independen. Untuk hasil uji coba konsistensi rekomendasi AHP dapat dilihat pada Tabel 6.7

**Tabel 6.7 Hasil Uji Coba Konsistensi Hasil Rekomendasi AHP**

Data	Hasil Rekomendasi			Hasil Sebenarnya
	Model 1	Model 2	Model 3	
1	Teknik Informatika	Teknik Kimia	Teknik Pengairan	Teknik Sipil
2	Teknik Mesin	Teknik Kimia	Teknik Industri	Arsitektur
3	Kedokteran	Gizi	Kedokteran	Kedokteran Gigi
4	Gizi	Gizi	Kedokteran	Gizi
5	Statistika	Matematika	Statistika	Matematika
6	Tenik Kelautan	Teknik Mesin	Teknik Kelautan	Teknik Material dan Metalurgi
7	Teknik Informatika	Teknik Kelautan	Teknik Material dan Metalurgi	Teknik Mesin

8	Kedokteran	Kedokteran Gigi	Kedokteran	Kedokteran
9	Teknik Pengairan	Teknik Industri	Teknik Material dan Metalurgi	Teknik Industri
10	Statistika	Matematika	Statistika	Matematika

Dari Tabel 6.7 dapat dilihat, dari 10 kali percobaan terhadap ketiga model *AHP*, ditemui perbedaan di dalam memberikan rekomendasi. Terdapat 4 percobaan dimana ketiga model yang masing-masing memberikan rekomendasi berbeda, 6 percobaan dimana dua dari tiga model memberikan rekomendasi yang sama, dan tidak ada percobaan dimana ketiga model memberikan hasil rekomendasi yang sama. Dari percobaan ini dapat dikatakan, sistem masih belum bisa memberikan hasil rekomendasi yang sama untuk setiap modelnya, namun sistem umumnya memberikan dua hasil rekomendasi yang sama dari tiga modelnya, sehingga untuk memperoleh rekomendasi yang lebih kuat, *user* disarankan mencoba ketiga model *AHP* didalam melakukan perhitungan dan melihat hasil yang paling dominan dari tiga hasil rekomendasi jurusan yang diberikan.

Untuk akurasi ketepatan hasil rekomendasi dengan hasil sebenarnya dari data uji, dari 10 kali percobaan, model 1 memiliki 2 hasil rekomendasi yang sama dengan hasil sebenarnya, sementara model 2 memiliki 4 hasil rekomendasi yang sama dengan hasil sebenarnya, dan model 3 memiliki 1 memiliki hasil rekomendasi yang sama dengan hasil sebenarnya. Dari hasil tersebut dapat dikatakan bahwa model 2 *AHP* memiliki hasil rekomendasi yang lebih baik dibandingkan dengan model 1 dan model 3.



## BAB 7 PENUTUP

Bab ini membahas kesimpulan dan saran dari penelitian sistem klasifikasi jurusan kuliah untuk pelajar SMA menggunakan algoritma *AHP-Naïve Bayes*

### 7.1 Kesimpulan

Berdasarkan hasil penelitian mengenai implementasi sistem klasifikasi jurusan kuliah untuk pelajar SMA menggunakan algoritma *AHP-Naïve Bayes*, didapatkan beberapa kesimpulan, yaitu :

1. Metode *AHP* bersama metode *Naïve Bayes* dapat diimplementasikan untuk sistem klasifikasi jurusan kuliah untuk pelajar SMA. Hasil dari sistem adalah jurusan yang direkomendasikan berdasarkan nilai dari calon mahasiswa. Di dalam sistem ini, data diklasifikasikan dahulu ke dalam sebuah fakultas dengan menggunakan metode naive bayes, kemudian data yang memiliki kelas diterima di suatu fakultas akan diproses oleh metode *AHP* untuk pemberian jurusan. Proses pemberian jurusan melibatkan proses pembuatan matriks perbandingan berpasangan di dalam metode *AHP* untuk data nilai calon mahasiswa di dalam perhitungannya yang memerlukan konsistensi. Sistem ini memiliki 3 model *AHP* di dalam perhitungannya. Dari hasil pengujian konsistensi hasil, didapatkan kesimpulan bahwa hasil rekomendasi jurusan yang lebih kuat didapatkan ketika 2 model memiliki hasil rekomendasi yang sama.
2. Implementasi sistem dimulai dari pemilihan fakultas dengan menggunakan algoritma *naïve bayes*, dengan akurasi tertinggi ketika sistem menggunakan data latih berjumlah 90% dari total data latih dengan akurasi 95 %. Namun akurasi ini didapatkan karena data uji berasal dari data dengan kelas terbanyak. Hal ini membuktikan tingkat akurasi sistem sangat dipengaruhi oleh jumlah data latih yang didapatkan. Akurasi yang didapat akan berbeda bila jumlah data latih yang dilibatkan berbeda.

### 7.2 Saran

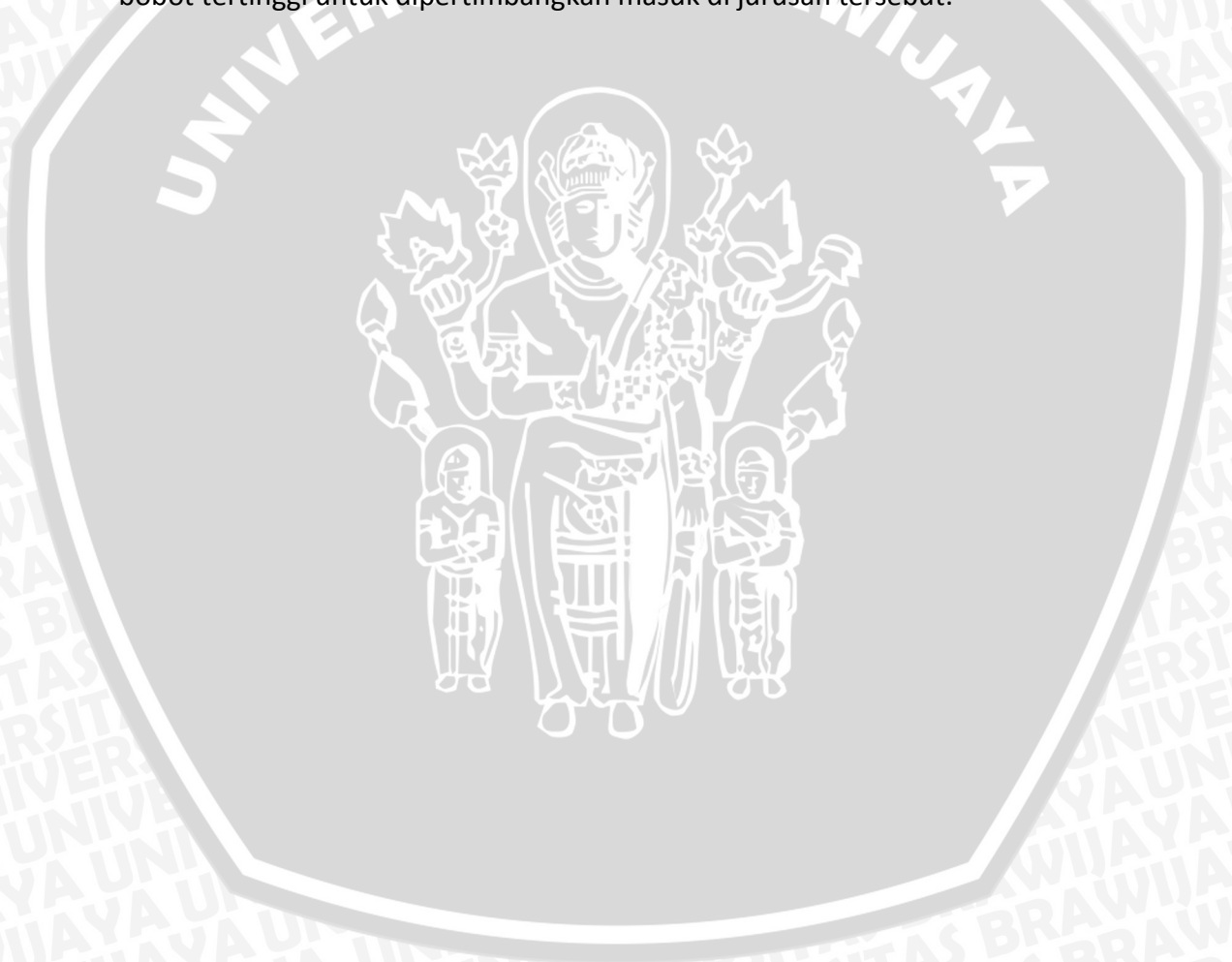
Berdasarkan kesimpulan yang telah didapat, ada beberapa saran untuk pembaca jika berniat untuk melakukan penelitian lebih lanjut

1. Data latih sangat penting untuk penelitian ini, tidak hanya sekedar jumlah, namun juga proporsi jumlah kelas data, yaitu kelas “diterima” dan “tidak”, sehingga pada penelitian berikutnya dapat dilakukan pengumpulan data yang lebih terstruktur lagi.
2. Dalam penelitian selanjutnya disarankan pengumpulan data dilakukan di satu tempat atau di satu SMA dikarenakan parameter rekomendasi yang diperhitungkan hanyalah berdasarkan nilai. Jika pengumpulan data berasal dari berbagai SMA, maka perlu diperhitungkan juga indeks sekolah dan beberapa faktor lainnya. Dan jika memungkinkan, data yang dikumpulkan



adalah data yang memiliki universitas tujuan yang sama, sehingga hasil yang didapat lebih baik lagi.

3. Dalam penelitian ini tidak memperhitungkan parameter psikologi dari calon mahasiswa, sehingga dalam penelitian berikutnya dapat ditambahkan perhitungan dengan menambahkan parameter psikologi anak untuk mendapatkan hasil rekomendasi yang lebih baik lagi.
4. Untuk penelitian selanjutnya, dapat dikembangkan model AHP dimana *goal* dari sistem adalah fakultas atau jurusan dari universitas, sehingga memudahkan fakultas/jurusan untuk memilih mahasiswa. Sistem yang dikembangkan memiliki 2 level dimana sistem akan memproses data nilai bobot untuk tiap mata pelajaran yang berlaku di jurusan tersebut, kemudian hasil pembobotan mata pelajaran tersebut akan dikalikan dengan data nilai para calon mahasiswa sehingga didapatkan calon mahasiswa yang memiliki bobot tertinggi untuk dipertimbangkan masuk di jurusan tersebut.



## DAFTAR PUSTAKA

- Chan, et al. 2013. *An Integration of AHP Approach and Bayes Classification Algorithm in Supplier Selection*. Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong, China
- Druzdzel, Flynn. 2002. *Decision Support Systems*. Pittsburgh. University of Pittsburgh
- Deorah, Sourabh, Sridharan, Srivatsan, Goel, Shivani. 2010. *SAES- Expert System for Advising Academic Major*. Computer Science and Engineering Department Thapar University
- Pavani, Sirigiri, Sharma, Lokesh Kumar, S., Hota H., 2013. *A Group Expert Evaluation for Teachers by Integrating Fuzzy AHP and TOPSIS Models*. IEEE, 978-1-4799-1626-9.
- Slim, Ahmad, et al, 2011. *Predicting Student Success Based on Prior Performance*. USA: Department of Electrical and Computer Engineering.
- Shunseng, Sun, Qin, Zhang, Min, Zhang, 2011. *Evaluation of Building Materials Logistics Provider Based on FAHP-TOPSIS*. International Conference on Uncertainty Reasoning and Knowledge Engineering.
- Saaty, T.L., 1994, *Fundamental Of Decision Making and Priority Theory With The Analytic Hierarchy Process*, University of Pittsburgh, RWS publication.
- Manap, Tahir, Abdullah, 2012. *Anomalous Gait Detection using Naive Bayes Classifier*, IEEE Symposium on Industrial Electronics and Applications.
- Mohamad, Ardyan. 2014. <http://www.merdeka.com/uang/baru-20-persen-penduduk-kuliah-indonesia-sulit-jadi-negara-maju.html>, diakses 9 September 2015
- Novak, T., & Weiss, D. (1988). What's all this talk about enrollment management? *The Admissions Strategist*, 3: 1-5
- Universitas Brawijaya, 2011. Fakultas dan Program, <http://ub.ac.id/akademik/fakultas-id>, diakses pada 3 September 2015.
- Yudatama, Uky, Sarno, Riyanarto, 2015. *Evaluation Maturity Index And Risk Management For IT Governance Using Fuzzy AHP And Fuzzy TOPSIS*. International Seminar on Intelligent Technology and Its Application.
- Zhu, Fangjuan, Liu, Ye, 2011. *Assessment of the information disclosure level about government website through AHP-TOPSIS method*. China. College of Management, Tianjin polytechnic university.

## LAMPIRAN 1 SURAT PERMOHONAN DATA SKRIPSI



KEMENTERIAN RISET, TEKNOLOGI, DAN PENDIDIKAN TINGGI  
UNIVERSITAS BRAWIJAYA

PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

Gedung A PTIIK Lt. 1, Jl. Veteran No.8, Malang, 65145, Indonesia

Telp. : +62-341-577911; Fax : +62-341-577911

http://ptiik.ub.ac.id E-mail : ptiik@ub.ac.id

Nomor : 2817 /UN10.36/AK/2016

Perihal : *Permohonan data skripsi*

07 APR 2016

Yth. Kepala Dinas Pendidikan Kota Malang  
Jl. Veteran No. 19  
Malang

Untuk mendukung penyelesaian skripsi mahasiswa berikut :

Nama : Davin Ega Prasetya  
NIM : 125150200111058  
Judul Skripsi : Sistem Pengklasifikasi Penentuan Jurusan Kuliah Untuk Pelajar SMA Dengan Algoritma AHP-Naïve Bayes  
Dosen Pembimbing : 1. Imam Cholissodin, S.Si.,M.Kom  
2. Agus Wahyu Widodo, S.T., M.Cs.  
Prodi : Informatika / Ilmu Komputer

Guna memperoleh data untuk skripsi mahasiswa tersebut di instansi saudara, jenis data yang diperlukan dan rencana waktu pelaksanaan adalah :

Data : Surat Permohonan izin kegiatan penelitian di SMAN 1 Malang  
Waktu : 08 April 2016 - 08 Mei 2016

Atas perhatian dan kerjasamanya kami ucapkan terimakasih.

Mengetahui,  
a.n. Ketua Program  
Kepala Tata Usaha PTIIK,

Dosen Pembimbing I



Dra. Endang Budha Mahadyawati, M.AP.  
NIK/ NIP. 19561213 19820 2 2001

Imam Cholissodin, S.Si.,M.Kom  
NIK/ NIP. 201201 850719 1 001

**Tembusan Kepada Yth:**

1. Ketua Program Studi Informatika / Ilmu Komputer
2. Mahasiswa yang bersangkutan



## LAMPIRAN 2 SURAT REKOMENDASI DINAS PENDIDIKAN



### PEMERINTAH KOTA MALANG DINAS PENDIDIKAN

Jl. Veteran No. 19 Telp. (0341) 560946, Fax. (0341) 551333  
 Website : <http://diknas.malangkota.go.id> | Email : [disdik\\_mlg@yahoo.co.id](mailto:disdik_mlg@yahoo.co.id)  
 Kode POS : Malang 65145

### REKOMENDASI

Nomor : 074 / 1335/ 35.73.307 / 2016

Menunjuk surat dari Ketua Program Kepala Tata Usaha PTIIK Program Teknologi Informasi Dan Ilmu Komputer Universitas Brawijaya Malang tanggal 07 April 2016 Nomor 2817/UN10.36/AK/2016 Perihal :Permohonan Data Skripsi, maka dengan ini kami berikan ijin untuk melaksanakan kegiatan dimaksud kepada:

1. Nama : Davin Ega Prasetya
2. NIM : 125150200111058
3. Jenjang : S1
4. Prodi. / Jurusan : Informatika / Ilmu Komputer
5. Tempat Pelaksanaan : SMA Negeri 1 Malang
6. Waktu Pelaksanaan : April s.d Mei 2016
7. Judul : Sistem Pengklasifikasi Penentuan Jurusan Kuliah Untuk Pelajar SMA Dengan Algoritma AHP-Naive Bayes

DenganKetentuan :

1. Dikoordinasikan sebaik – baiknya dengan Kepala SKPD / Sekolah;
2. Tidak Mengganggu proses belajar – mengajar;
3. Berlaku selama tidak menyimpang dari peraturan;
4. Selesai melaksanakan penelitian / Observasi / KKL / KKN, wajib menyampaikan laporan kepada Kepala Dinas Pendidikan Kota Malang.

Demikian untuk dilaksanakan sebagaimana mestinya.

Malang, 07 April 2016



Tembusan :

1. Kepala SMA Negeri 1 Malang
2. Ketua Program Studi Informatika / Ilmu Komputer
3. Yang bersangkutan



## LAMPIRAN 3 DATA NILAI PELAJAR SMA

### Kelas Akselerasi

- Sem 3

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	GE O	EK O	SO S	SN B	PJO	TI K	KB A
1	84	83	84	83	83	79	85	86	78	84	87	80	81	84	83	88
2	89	90	82	84	84	94	86	91	87	89	85	86	82	87	84	85
3	87	88	87	85	91	83	86	93	85	89	90	84	92	85	88	92
4	90	89	89	82	82	83	84	89	83	89	87	80	82	85	88	91
5	85	87	82	84	84	90	87	92	78	88	86	80	80	86	89	86
6	86	89	86	85	87	88	85	91	86	89	88	83	81	85	87	92
7	86	84	88	84	84	78	84	80	82	82	90	80	78	85	85	90
8	86	87	85	83	82	82	85	90	78	89	93	80	81	84	87	89
9	87	84	84	87	84	78	84	84	82	85	91	80	83	85	90	88
10	84	87	86	84	83	83	85	90	86	89	86	81	78	85	86	85
11	89	86	86	86	88	83	86	86	83	89	87	80	78	85	87	86
12	87	85	84	85	82	84	83	87	80	91	92	80	78	84	86	88
13	87	86	90	87	92	93	87	95	90	89	90	84	79	86	86	92
14	89	91	85	86	85	86	84	91	85	89	92	90	83	85	89	89
15	89	85	86	84	83	87	85	90	82	88	87	84	86	85	85	92
16	86	88	84	84	93	80	85	89	80	88	91	81	85	85	85	90
17	88	89	86	86	95	82	88	93	88	87	91	80	81	86	89	90

- Sem 4

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	GE O	EK O	SO S	SN B	PJO	TI K	KB A
1	84	85	83	82	80	80	82	87	80	78	85	87	82	84	85	83
2	89	87	81	85	93	88	84	88	80	87	87	87	83	89	87	81
3	92	87	87	85	90	93	86	93	86	86	85	88	94	92	87	87
4	90	89	87	82	90	82	83	87	82	83	85	88	86	90	89	87
5	87	88	83	85	87	80	85	89	84	85	86	89	82	87	88	83
6	86	84	83	84	88	90	85	89	80	82	85	87	85	86	84	83
7	80	81	83	84	81	80	82	80	80	82	85	88	84	80	81	83
8	85	82	83	82	80	80	83	88	80	80	84	88	85	85	82	83
9	84	90	85	85	86	80	81	85	80	81	85	90	84	84	90	85
10	90	90	83	85	80	80	83	86	87	82	85	87	81	90	90	83
11	90	83	86	85	87	90	86	90	80	83	85	88	83	90	83	86
12	80	82	87	83	80	82	82	86	80	80	84	87	82	80	82	87
13	86	85	85	86	93	93	85	90	80	84	86	87	86	86	85	85
14	91	92	84	85	88	80	84	88	80	80	85	89	91	91	92	84

15	84	86	87	85	87	80	85	88	82	87	85	88	85	84	86	87
16	88	89	85	82	85	85	83	90	88	80	85	86	82	88	89	85
17	84	86	84	85	90	87	84	87	90	83	86	89	89	84	86	84

- Sem 5

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	GE O	EK O	SO S	SN B	PJO	TI K	KB A
1	80	86	86	82	86	80	81	88	78	78	79	83	82	80	86	86
2	94	89	90	84	98	85	91	88	90	78	81	91	85	94	89	90
3	94	89	90	83	93	92	93	92	88	80	87	87	94	94	89	90
4	92	88	90	82	88	85	94	92	87	78	91	89	89	92	88	90
5	87	89	90	84	88	79	87	84	90	78	87	88	85	87	89	90
6	86	86	86	83	84	86	86	91	81	82	81	86	89	86	86	86
7	80	86	86	82	81	78	82	84	78	78	79	87	83	80	86	86
8	87	85	88	83	81	81	83	90	87	78	83	88	86	87	85	88
9	85	89	86	86	83	81	87	88	80	78	96	89	82	85	89	86
10	84	89	86	88	86	85	86	84	89	78	89	86	82	84	89	86
11	88	88	86	89	90	92	88	84	90	78	80	88	80	88	88	86
12	85	85	90	87	89	86	89	88	90	78	81	88	86	85	85	90
13	88	87	90	89	86	87	92	89	87	78	80	88	89	88	87	90
14	95	92	90	88	90	86	93	88	85	78	89	91	89	95	92	90
15	93	89	90	88	82	86	93	89	87	80	87	87	90	93	89	90
16	93	89	86	83	94	87	88	91	84	78	80	86	82	93	89	86
17	89	89	90	86	92	90	90	90	86	78	81	88	86	89	89	90

Kelas IPA 1

- Sem 3

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	83	86	86	80	82	80	83	88	81	85	77	82	78
2	87	86	87	80	81	80	79	91	82	89	78	84	92
3	89	89	84	85	94	82	82	89	82	86	77	83	84
4	87	81	81	88	88	79	78	87	83	79	79	80	80
5	89	86	82	78	91	84	79	84	84	88	77	84	97
6	83	90	84	80	90	78	78	86	82	85	76	82	93
7	87	84	83	83	90	78	78	84	82	80	78	83	69
8	85	88	82	83	91	81	78	84	82	83	79	85	86
9	88	88	89	80	95	97	83	92	87	88	75	83	92
10	80	82	80	81	80	79	78	80	79	81	75	78	78



11	84	84	87	81	92	79	81	86	82	85	78	86	85
12	80	88	81	84	88	83	80	88	78	86	75	79	93
13	83	89	80	81	93	88	83	86	85	82	77	86	88
14	86	92	81	86	93	95	82	89	86	85	80	85	88
15	88	88	83	80	97	91	84	93	84	90	75	83	96
16	87	85	84	80	99	83	80	89	81	93	79	84	95
17	84	87	84	80	86	90	79	85	81	90	76	83	92
18	90	88	86	81	95	97	82	89	80	83	78	84	90
19	87	89	86	81	94	82	79	88	80	84	79	81	79
20	93	88	86	84	91	80	82	90	82	91	80	85	93
21	88	87	81	78	93	86	80	87	80	86	83	87	96
22	86	90	86	84	96	85	82	90	83	88	84	82	88
23	89	89	84	83	92	95	83	94	87	91	83	88	93
24	84	91	86	86	90	95	81	89	85	83	87	80	85
25	80	84	82	84	88	91	78	84	78	85	79	82	85
26	84	84	83	83	93	83	78	86	85	86	78	87	91
27	86	88	84	87	84	96	78	89	86	84	84	86	92
28	82	88	86	84	86	90	81	88	80	85	78	86	93
29	87	83	81	78	82	80	78	84	80	86	75	81	89
30	89	84	85	80	82	86	79	85	81	88	77	82	97
31	93	87	86	82	96	85	81	92	82	89	83	86	96
32	88	91	86	82	96	88	78	87	82	90	75	85	90
33	91	91	88	89	93	90	82	90	85	89	77	88	93
34	86	91	87	85	96	80	81	87	84	85	87	84	89
35	91	94	84	83	94	79	80	89	86	86	82	78	86

- Sem 4

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	84	85	89	81	84	80	82	89	85	88	84	86	79
2	88	88	89	83	83	90	82	90	89	88	84	92	88
3	86	86	88	85	95	80	81	87	88	84	85	91	79
4	84	82	80	88	89	79	79	82	79	79	81	90	79
5	84	90	88	85	91	86	79	92	90	90	80	82	90
6	84	87	87	87	90	79	79	81	84	86	78	89	97
7	84	84	80	82	90	79	84	81	84	46	78	85	78
8	84	86	81	84	91	82	80	86	88	81	83	85	85
9	92	90	90	85	95	91	86	94	86	88	83	89	98



10	80	85	81	84	82	79	79	80	79	82	76	84	79
11	84	85	80	83	92	84	79	88	83	88	76	92	79
12	87	86	88	84	90	90	84	90	84	87	78	88	91
13	84	85	86	85	95	80	80	86	88	82	75	84	80
14	87	87	87	84	93	88	80	91	89	85	80	88	90
15	89	90	88	89	93	90	84	92	87	88	82	86	92
16	93	89	88	80	99	90	83	94	86	90	81	90	92
17	88	90	82	80	89	91	79	84	86	88	79	87	92
18	89	87	88	82	95	91	79	83	84	86	84	88	91
19	84	82	79	81	94	79	80	80	78	83	81	86	78
20	91	90	89	88	91	82	81	88	90	89	84	86	99
21	85	87	86	80	93	86	82	87	81	87	82	85	88
22	87	85	83	83	96	85	82	88	85	84	87	85	88
23	91	90	89	87	92	90	87	95	85	91	85	88	95
24	89	88	85	93	90	84	84	91	86	85	87	86	88
25	88	85	85	91	89	80	79	90	86	86	83	84	84
26	90	86	87	91	93	80	81	89	85	88	83	90	89
27	93	86	87	91	84	85	80	90	85	84	85	91	87
28	85	89	85	90	86	85	82	91	89	85	83	89	89
29	91	86	87	80	86	85	79	88	83	85	85	87	87
30	87	87	85	82	86	90	82	89	84	87	75	86	91
31	93	89	84	88	96	86	86	91	88	88	85	91	92
32	91	89	84	88	96	80	82	90	91	88	82	90	87
33	93	89	83	91	93	85	84	88	84	86	80	92	92
34	93	86	86	92	96	81	79	86	86	87	86	87	87
35	92	86	85	88	94	82	79	85	88	88	82	84	85

- Sem 5

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	87	88	84	84	94	86	86	87	86	89	79	90	83
2	92	88	92	83	90	89	82	88	87	89	83	87	92
3	78	88	85	83	85	82	83	82	87	90	78	87	83
4	85	85	81	81	80	84	80	80	78	85	82	83	80
5	92	90	88	80	92	89	87	86	83	89	84	90	88
6	90	88	89	83	92	83	82	83	85	89	81	86	89
7	78	86	82	80	83	80	82	80	80	85	81	82	79
8	85	88	84	83	87	84	85	85	88	86	83	86	86





9	94	90	99	89	95	97	85	94	88	90	85	88	99
10	84	85	90	81	83	85	82	81	81	86	80	86	90
11	88	89	83	81	93	88	82	88	85	88	79	88	83
12	87	88	82	90	90	90	85	89	86	89	80	93	90
13	78	86	91	83	86	86	85	89	87	82	84	83	91
14	81	88	86	81	94	93	80	89	85	78	80	82	83
15	90	90	94	85	98	94	84	90	88	88	82	91	94
16	92	88	96	84	98	93	86	94	88	91	82	91	96
17	90	88	90	84	89	92	86	87	90	91	78	87	85
18	92	88	89	83	96	97	85	85	86	89	78	87	93
19	84	88	82	82	81	83	80	84	80	88	82	83	75
20	90	90	89	83	97	89	87	89	90	88	83	88	99
21	89	88	84	86	91	86	86	86	87	89	81	88	88
22	90	86	85	83	90	91	83	88	85	87	86	86	88
23	93	90	88	86	99	95	88	92	89	91	84	90	95
24	88	87	86	81	94	92	85	90	87	89	88	83	93
25	86	87	85	81	89	88	85	91	85	87	87	87	83
26	90	86	87	81	95	89	85	91	87	88	85	88	91
27	83	88	88	86	88	92	82	92	86	87	84	87	86
28	90	90	85	86	88	81	84	87	88	88	81	85	89
29	96	87	87	81	87	82	85	83	85	89	79	88	90
30	93	86	85	83	91	90	85	90	86	89	84	88	87
31	93	90	87	85	96	89	85	90	88	89	82	87	96
32	95	90	89	86	96	89	85	89	89	90	83	91	88
33	95	90	92	84	99	90	85	91	88	89	85	88	93
34	89	88	87	82	90	86	84	87	86	90	84	84	93
35	89	88	88	84	93	85	84	90	88	88	83	88	85

Kelas IPA 2

- Sem 3

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	89	85	88	88	87	90	78	85	83	86	79	86	92
2	88	84	83	80	82	83	78	81	80	87	78	79	82
3	84	84	84	79	82	85	80	85	80	87	75	81	88
4	85	84	84	78	90	85	80	81	78	85	85	81	85
5	91	86	83	79	83	83	81	88	80	86	77	82	81
6	90	89	81	86	95	90	79	88	84	86	81	84	96



7	80	84	86	80	90	86	78	83	78	83	77	82	97
8	81	85	85	84	80	84	78	85	81	86	84	84	83
9	85	85	81	84	87	80	80	88	81	88	82	84	92
10	85	87	80	79	94	85	78	93	82	85	84	87	81
11	88	83	88	83	82	82	80	84	84	88	79	84	86
12	88	85	84	86	91	94	79	81	86	83	77	84	78
13	90	84	87	85	92	80	79	88	86	89	81	86	96
14	91	86	86	79	96	81	81	95	87	90	83	87	96
15	90	88	84	79	93	88	81	92	87	86	82	86	90
16	85	89	84	76	97	80	79	84	85	40	80	87	84
17	88	85	85	79	95	92	79	81	81	88	80	81	81
18	93	85	89	80	86	90	80	89	82	86	77	83	96
19	87	85	89	79	85	85	80	81	80	88	76	78	88
20	85	85	90	80	87	81	80	85	83	85	76	78	89
21	84	86	82	76	82	83	82	86	84	41	82	84	78
22	86	84	83	80	88	78	78	87	82	88	75	80	88
23	89	86	88	82	92	80	79	82	82	85	75	78	91
24	87	83	81	79	94	82	79	86	84	82	83	86	87
25	90	85	86	79	85	81	79	87	85	85	83	87	84
26	88	86	82	84	83	94	79	86	84	83	78	83	89
27	89	83	83	86	91	92	78	85	85	81	78	78	89
28	93	84	85	81	93	86	81	86	84	85	80	79	91
29	89	86	82	79	92	85	80	91	83	85	78	86	88
30	85	86	87	80	93	81	87	95	83	88	79	82	78
31	88	85	84	80	91	81	80	81	82	87	78	85	87
32	89	84	86	80	93	85	79	90	81	89	84	87	83
33	89	84	83	79	95	88	80	81	85	84	81	86	89
34	82	88	87	76	91	93	78	83	86	86	84	81	85

- Sem 4

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	86	85	90	84	87	84	79	80	80	87	82	88	93
2	88	85	86	82	83	81	79	84	84	89	83	85	87
3	87	86	84	81	83	79	79	90	80	87	79	84	90
4	88	85	90	78	90	85	79	80	80	88	84	86	88
5	91	87	87	84	84	88	86	86	89	86	84	88	88
6	84	89	94	78	95	90	81	86	83	88	80	89	86



7	83	84	90	82	90	81	79	82	79	84	77	86	97
8	92	85	88	78	83	80	80	80	81	88	84	86	84
9	86	84	84	79	90	80	81	80	82	88	88	79	83
10	86	87	90	78	95	85	79	86	86	88	88	88	81
11	87	85	86	81	87	81	79	87	80	89	83	89	79
12	87	86	86	79	93	82	83	85	84	87	85	87	80
13	93	88	90	81	92	83	83	92	88	89	85	89	90
14	95	88	93	81	96	87	83	96	89	90	86	89	94
15	94	89	88	79	93	85	81	91	83	88	87	85	84
16	91	88	87	78	97	84	84	88	84	84	82	82	78
17	84	85	83	80	95	89	79	80	80	90	86	81	82
18	92	87	90	81	87	89	82	88	80	89	81	84	92
19	87	85	84	78	85	80	82	84	85	88	80	84	87
20	87	86	89	79	87	80	80	90	90	88	83	84	82
21	89	85	81	78	82	81	84	81	90	76	86	78	83
22	88	86	90	84	89	80	80	93	84	89	78	83	83
23	86	84	87	80	92	79	79	80	86	87	78	82	85
24	90	85	90	81	94	79	81	84	88	85	87	86	84
25	87	85	86	79	85	83	81	87	86	85	88	89	86
26	85	85	84	78	85	90	82	82	80	86	77	88	80
27	84	85	88	86	92	80	82	83	81	84	79	84	84
28	84	85	86	81	93	79	79	81	80	86	82	88	86
29	85	87	88	79	92	90	79	91	90	88	90	92	83
30	88	87	89	82	93	83	80	93	90	89	79	87	89
31	89	85	83	81	91	80	79	87	88	88	83	85	88
32	88	87	92	87	93	92	80	90	90	90	83	90	93
33	84	85	85	79	95	83	79	87	85	88	80	91	85
34	84	85	80	79	91	78	79	81	80	86	85	86	79

- Sem 5

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	85	86	88	88	84	87	84	83	84	84	78	83	97
2	86	86	86	87	96	83	83	81	80	88	79	86	92
3	85	86	85	83	85	80	84	86	80	83	79	87	83
4	87	85	87	81	91	89	83	90	85	87	81	85	84
5	88	87	91	90	92	83	86	86	85	89	78	87	91
6	85	85	86	85	95	90	85	87	79	89	78	83	89



7	78	88	90	82	81	85	84	83	84	84	79	85	97
8	86	86	86	84	88	81	82	88	86	88	81	87	91
9	78	86	85	81	89	88	80	84	82	86	84	85	84
10	84	87	90	83	94	90	83	89	82	89	83	91	86
11	83	85	86	84	92	86	83	85	84	87	82	85	93
12	81	87	88	82	85	88	80	86	80	86	82	83	81
13	92	88	93	84	93	92	85	87	82	90	81	89	90
14	88	88	88	88	92	92	82	89	84	90	83	83	94
15	88	89	87	84	94	93	85	87	86	88	84	85	89
16	78	87	89	80	82	90	80	84	82	85	82	80	80
17	82	85	85	81	90	90	85	80	81	88	82	87	78
18	94	88	85	84	92	90	85	88	83	89	78	87	90
19	83	85	87	82	91	91	86	84	84	90	79	85	81
20	88	86	89	81	91	86	85	83	87	88	79	88	89
21	78	86	84	80	82	87	80	84	80	82	81	79	89
22	85	87	82	84	85	85	84	84	83	88	82	88	82
23	89	85	84	80	85	79	84	80	86	79	80	87	82
24	85	87	82	81	93	90	85	84	87	85	85	81	85
25	89	88	82	83	91	89	84	90	84	90	85	90	79
26	85	87	84	83	90	90	83	90	84	87	76	83	86
27	88	85	85	81	88	80	82	85	84	84	81	79	86
28	88	85	90	84	85	88	85	83	84	89	80	86	88
29	88	87	90	81	93	90	86	90	86	87	80	84	86
30	89	87	90	86	90	86	84	90	84	88	80	84	81
31	85	85	86	83	87	84	84	80	82	87	80	86	88
32	91	89	90	90	96	94	85	89	86	90	83	92	95
33	87	86	82	85	91	91	80	87	82	90	79	82	91
34	78	87	86	81	80	90	84	80	82	88	82	88	85

Kelas IPA 3

- Sem 3

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	84	91	88	83	89	80	78	83	82	83	76	86	86
2	87	86	83	81	88	78	80	81	78	83	83	80	81
3	82	84	80	81	94	80	78	82	78	82	79	82	90
4	87	81	84	90	98	93	78	86	80	86	79	89	80
5	89	89	82	84	81	79	83	92	83	85	77	78	89



6	90	88	84	84	95	93	79	85	85	84	84	80	87
7	90	87	83	88	95	89	78	88	80	83	80	86	88
8	90	83	83	88	85	79	78	81	84	84	76	79	86
9	91	84	80	79	85	78	78	86	82	86	79	80	87
10	84	85	84	81	87	84	79	86	84	86	77	90	78
11	84	84	85	82	86	85	78	87	87	86	78	81	88
12	80	83	81	82	80	78	78	82	78	83	75	80	87
13	85	87	83	87	90	85	80	93	86	86	75	80	90
14	80	82	83	84	87	80	78	82	82	84	75	80	83
15	83	85	85	90	90	85	79	84	84	85	75	83	93
16	80	85	81	90	85	88	78	82	78	39	77	82	78
17	92	90	89	87	92	80	80	85	82	84	81	82	78
18	86	88	83	91	80	78	78	80	85	86	75	79	92
19	85	84	83	88	85	78	79	85	85	86	83	82	78
20	86	82	84	85	90	80	80	84	80	86	76	82	89
21	86	85	83	87	85	82	80	84	83	86	85	80	89
22	85	87	84	79	83	78	78	84	81	85	80	80	89
23	84	82	80	76	80	78	78	82	79	84	75	76	89
24	84	83	82	83	89	92	79	87	80	88	84	84	92
25	85	84	81	80	85	78	78	83	78	84	74	82	78
26	85	88	86	88	87	81	78	84	83	88	78	86	92
27	85	89	88	85	84	79	80	83	84	84	75	84	97
28	84	86	90	87	82	81	80	89	89	87	79	83	89
29	85	88	88	87	89	78	78	89	86	86	78	80	78
30	86	83	81	86	91	81	82	85	80	85	81	84	81
31	86	87	86	87	83	79	78	87	85	87	77	80	82
32	85	84	90	83	86	78	80	87	86	90	77	82	88
33	87	84	84	82	86	78	80	83	85	80	77	82	78
34	89	83	82	84	80	80	83	86	85	86	78	78	86
35	89	91	86	81	85	79	79	95	88	85	84	82	80

- Sem 4

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	85	86	89	81	89	80	80	81	83	83	80	83	81
2	90	86	83	79	88	79	81	80	80	84	84	88	81
3	81	84	83	81	94	80	80	83	84	84	85	87	85
4	87	86	82	88	98	90	81	84	89	89	80	92	80



5	84	86	80	83	81	83	83	91	86	86	80	86	84
6	91	85	86	83	95	88	87	84	85	85	83	83	84
7	92	87	84	82	95	84	87	86	85	85	82	88	85
8	84	89	82	78	89	79	80	82	84	84	78	84	89
9	84	91	84	78	85	79	80	88	87	87	82	87	87
10	87	88	84	78	87	79	80	87	88	88	82	88	81
11	85	88	80	78	86	79	80	85	86	86	81	86	85
12	84	85	81	78	80	79	80	82	83	83	77	84	80
13	85	86	83	82	91	89	83	94	89	89	79	85	86
14	87	86	85	80	87	79	80	89	88	87	76	81	85
15	86	86	85	84	90	81	80	86	88	88	80	85	86
16	87	84	82	80	90	78	80	81	84	81	82	80	79
17	89	87	85	81	92	81	81	83	87	87	84	85	83
18	90	87	88	85	83	82	80	89	86	86	78	88	82
19	88	85	85	84	86	82	82	81	86	86	86	87	84
20	93	85	89	82	90	84	82	85	87	87	83	90	83
21	88	91	84	85	86	81	80	84	81	89	82	87	85
22	91	91	87	81	85	82	80	87	88	88	80	86	82
23	88	85	80	78	83	79	80	81	84	83	80	84	83
24	91	86	87	78	90	83	80	83	90	90	84	90	84
25	86	86	80	78	88	80	80	81	87	87	78	90	78
26	86	86	85	85	87	83	80	87	88	88	80	88	90
27	88	87	87	82	84	81	80	84	85	85	79	84	88
28	91	86	87	82	84	84	86	85	88	88	75	86	86
29	88	86	86	88	89	85	81	90	88	88	81	89	85
30	86	84	81	85	91	80	85	84	84	84	83	86	86
31	89	85	86	80	83	81	80	86	90	90	86	87	82
32	84	84	84	85	86	80	80	87	90	90	81	89	92
33	80	85	84	79	86	82	80	80	84	84	80	87	79
34	84	84	82	79	82	86	80	81	89	89	75	90	90
35	90	88	89	79	85	84	89	93	90	90	87	86	90

- Sem 5

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	79	87	88	83	81	80	84	87	83	82	78	87	91
2	80	87	86	82	81	91	82	87	84	82	81	88	84
3	84	85	83	83	84	89	83	84	82	88	84	84	87



4	95	86	85	85	93	95	84	88	82	89	80	91	83
5	83	86	83	82	84	81	85	87	80	88	78	86	80
6	82	86	90	85	85	90	83	86	80	81	85	85	84
7	80	88	85	82	82	90	80	85	86	86	82	88	87
8	88	89	85	82	81	86	84	87	89	88	78	86	87
9	90	90	84	82	94	80	84	86	85	89	80	87	88
10	89	88	90	82	80	81	83	85	85	89	80	86	85
11	78	86	82	82	81	80	80	78	81	79	78	80	90
12	79	85	84	83	80	81	84	84	79	88	78	86	80
13	91	87	88	84	90	95	86	94	84	89	82	85	80
14	83	86	86	84	84	81	82	81	84	87	78	88	84
15	89	86	84	86	82	90	83	80	86	89	78	84	90
16	78	86	84	81	80	80	80	80	80	78	82	80	81
17	89	88	90	87	84	88	86	84	87	87	88	87	83
18	91	87	90	88	83	85	84	83	85	82	78	86	86
19	91	85	85	86	80	82	83	87	86	88	81	88	80
20	90	85	85	86	91	80	84	89	86	88	84	89	92
21	80	89	85	87	80	87	83	89	84	88	81	86	85
22	82	91	90	85	85	81	84	85	84	89	79	85	87
23	78	87	86	83	80	81	80	80	84	82	79	87	83
24	89	86	84	86	81	91	82	85	84	88	83	92	90
25	81	87	84	82	80	87	84	80	82	84	82	83	81
26	87	87	86	85	85	87	83	85	82	87	82	86	90
27	93	86	87	85	85	90	83	83	85	89	81	84	89
28	89	88	88	84	85	88	84	85	86	90	82	86	94
29	86	88	87	89	85	81	83	87	87	89	82	86	81
30	83	86	85	86	84	82	85	87	84	84	78	87	83
31	87	85	86	87	85	87	86	88	82	90	86	84	85
32	88	88	87	89	90	81	86	89	86	89	81	88	94
33	78	85	85	82	80	80	80	85	79	87	78	84	80
34	86	85	83	85	80	80	85	83	80	88	78	86	92
35	88	87	85	87	83	88	85	90	88	90	82	86	81

Kelas IPA 4

- Sem 3

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	83	87	91	85	87	85	88	91	84	83	80	85	85

2	83	86	89	83	94	79	79	90	84	84	81	84	78
3	82	85	83	86	86	78	79	88	85	83	75	80	91
4	85	90	81	78	80	78	82	81	80	86	81	78	90
5	89	87	87	87	91	78	84	88	83	66	75	86	94
6	90	90	85	80	97	85	93	93	83	86	75	83	91
7	86	85	91	88	88	80	83	87	85	89	76	82	90
8	86	86	90	88	89	87	81	86	85	87	81	80	80
9	89	91	89	85	80	87	87	88	83	63	81	84	78
10	88	87	84	86	87	81	85	88	86	83	83	82	89
11	84	86	88	83	80	79	86	83	84	86	82	81	83
12	86	88	87	88	92	78	87	86	82	83	83	81	94
13	86	85	88	83	86	78	85	82	82	90	81	81	87
14	84	87	85	89	93	79	85	86	86	85	76	81	85
15	85	90	84	79	82	89	90	92	85	88	75	83	94
16	92	86	86	87	98	89	81	90	85	86	78	83	88
17	87	86	83	88	98	84	85	90	86	88	80	84	87
18	83	84	85	86	80	80	84	87	80	85	75	84	87
19	87	84	87	85	90	78	81	85	85	84	80	80	91
20	84	83	85	85	80	79	79	86	84	83	82	81	87
21	84	83	84	86	80	79	86	87	85	90	80	83	90
22	79	84	87	83	80	81	84	84	84	83	75	81	81
23	87	83	85	90	85	78	80	85	83	90	76	82	88
24	84	83	88	86	83	82	86	90	81	80	83	84	78
25	86	81	87	90	91	80	72	83	83	62	82	83	63
26	88	83	87	88	94	91	86	92	85	88	82	84	90
27	86	83	87	87	86	85	84	87	86	82	80	80	86
28	88	82	85	87	86	87	83	84	86	83	84	80	88
29	83	83	84	78	80	78	85	84	79	86	75	78	87
30	80	84	90	83	83	79	81	89	86	86	85	82	73
31	86	85	82	85	80	84	84	81	80	86	81	82	80
32	85	84	84	83	83	78	82	84	87	84	80	81	78
33	90	82	81	79	95	88	87	87	83	86	80	80	96
34	87	80	81	86	80	78	73	86	68	60	81	81	80
35	89	86	87	85	82	91	81	84	85	88	80	81	90

- Sem 4

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------





1	88	85	88	80	87	80	89	85	86	87	83	88	81
2	90	84	85	80	94	78	80	83	82	86	84	86	84
3	90	84	85	89	86	78	78	82	85	84	81	84	84
4	88	85	84	81	85	78	84	88	80	88	84	84	86
5	91	85	88	84	92	86	82	90	87	86	81	90	88
6	85	89	85	84	97	92	93	92	84	88	81	82	89
7	91	85	85	79	89	81	85	86	88	88	85	87	82
8	89	84	80	79	89	80	83	88	89	87	81	83	87
9	91	86	83	81	85	80	88	87	87	86	81	88	87
10	89	85	82	81	87	78	83	89	88	83	84	87	83
11	88	85	87	81	82	78	86	85	83	86	79	87	88
12	90	84	81	79	92	78	84	82	85	85	85	85	82
13	89	84	89	78	86	84	83	82	85	89	79	89	86
14	88	86	85	83	93	82	86	86	89	84	82	87	85
15	93	87	90	83	85	88	90	94	81	89	80	82	92
16	86	86	82	80	98	87	80	90	81	85	82	82	84
17	84	86	88	81	98	88	83	91	86	87	83	82	82
18	84	86	88	83	83	79	80	82	89	85	82	85	84
19	89	85	86	81	90	80	83	87	89	85	84	84	78
20	90	85	87	87	83	80	80	85	89	85	83	86	85
21	88	84	86	85	83	79	83	91	85	88	84	88	80
22	81	85	87	81	85	78	83	88	86	86	79	89	84
23	87	84	85	84	85	84	80	80	81	86	77	88	80
24	88	85	84	84	85	85	85	86	82	82	83	89	85
25	86	83	82	80	91	78	72	87	79	81	78	87	78
26	84	84	81	80	94	81	85	86	86	89	83	87	87
27	87	84	84	85	87	83	79	79	82	86	82	85	84
28	88	84	85	79	87	84	81	84	87	86	83	84	82
29	89	85	83	83	83	81	84	84	87	88	80	89	94
30	88	83	81	86	82	78	82	80	79	86	81	88	85
31	83	83	87	82	83	79	81	80	80	86	81	84	84
32	85	84	83	79	85	87	82	80	86	85	85	84	84
33	85	84	82	81	95	88	87	84	83	86	85	88	84
34	84	81	77	79	80	78	69	74	79	79	83	85	79
35	89	85	85	85	84	94	89	90	90	87	84	86	90



No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	91	88	90	83	88	85	85	90	80	88	82	89	91
2	85	86	89	82	86	84	80	88	80	88	85	89	87
3	84	85	80	81	82	80	80	78	78	78	61	83	88
4	89	87	86	86	85	86	85	88	87	89	85	86	88
5	85	89	90	86	89	84	80	85	90	89	83	89	95
6	89	89	82	82	92	92	85	90	88	90	84	90	91
7	89	88	88	88	90	80	85	87	85	90	82	88	86
8	91	86	90	81	87	84	84	90	82	89	82	90	90
9	90	88	90	82	87	87	85	91	88	88	81	84	90
10	93	86	91	83	89	86	82	90	85	88	86	90	84
11	85	87	85	82	83	83	83	86	84	88	83	84	86
12	85	86	89	82	85	80	82	88	88	88	85	88	88
13	81	86	87	82	83	80	82	82	84	83	81	87	78
14	93	87	89	83	87	83	84	88	84	88	79	88	85
15	90	89	86	86	95	82	85	85	88	88	81	84	96
16	89	86	89	80	89	83	80	88	86	81	81	84	89
17	87	87	84	83	91	80	86	92	84	91	82	90	81
18	83	88	86	83	88	80	84	88	82	88	83	85	86
19	89	86	88	82	85	81	83	87	82	81	81	83	87
20	88	85	91	81	89	88	85	88	86	88	81	88	85
21	92	86	85	84	90	87	84	88	86	91	78	83	96
22	87	87	82	86	91	80	84	88	88	90	82	92	88
23	90	85	82	83	85	80	83	88	88	89	81	85	84
24	86	86	86	86	89	85	85	89	87	79	83	86	82
25	78	86	83	81	84	80	80	86	78	78	80	79	79
26	88	86	85	83	89	87	84	87	83	90	82	85	88
27	81	85	83	80	81	80	82	88	82	88	81	86	81
28	92	85	85	86	88	81	84	83	82	87	80	88	86
29	92	85	84	85	85	82	84	82	85	89	79	89	93
30	78	86	85	81	87	82	80	85	78	78	79	81	78
31	89	86	86	83	84	80	84	87	82	89	80	86	88
32	87	85	86	82	84	81	82	85	82	88	83	86	86
33	88	86	86	85	91	86	84	89	85	89	83	89	89
34	78	86	85	80	83	80	80	84	78	78	79	58	90
35	90	86	88	83	83	88	85	87	84	90	84	85	90



## Kelas IPA 5

- Sem 3

No	AGM	KWN	IND	ING	MAT	FIS	KIM	BIO	SJR	SNB	PJO	TIK	KB A
1	84	83	86	81	89	92	92	90	81	88	83	80	89
2	80	87	80	86	84	89	78	86	80	83	77	80	80
3	81	87	82	88	80	92	83	86	84	85	81	80	87
4	92	87	88	80	94	94	85	85	87	83	80	80	92
5	92	83	84	82	80	92	80	82	88	78	77	80	87
6	84	88	81	81	91	80	73	81	86	58	81	80	78
7	83	89	85	85	91	82	81	87	84	82	83	80	87
8	80	91	80	80	82	81	79	83	85	81	79	78	78
9	83	87	80	81	91	82	82	82	84	83	82	81	94
10	85	85	80	76	80	78	71	78	78	85	75	81	71
11	88	88	80	77	93	86	79	85	78	79	77	80	94
12	88	89	80	81	95	86	89	93	83	85	81	81	92
13	94	89	89	78	95	89	83	86	84	79	87	80	88
14	88	89	86	81	89	81	81	88	83	88	79	78	90
15	89	88	87	85	86	82	80	87	84	87	83	80	82
16	81	87	82	80	83	82	79	80	80	65	83	81	67
17	88	87	80	82	83	80	81	87	86	63	85	81	86
18	89	89	88	85	86	81	84	85	70	89	78	80	84
19	82	86	80	83	84	82	88	82	78	84	75	80	78
20	88	88	88	86	95	88	86	90	85	85	82	85	94
21	85	92	81	82	92	85	81	86	84	85	84	82	87
22	87	83	86	80	83	82	79	86	82	80	85	79	80
23	88	83	86	84	80	80	86	85	82	84	78	84	89
24	84	84	80	84	85	80	80	81	80	78	79	78	89
25	89	83	84	81	92	92	95	98	84	86	77	85	89
26	90	88	84	85	81	84	86	95	86	89	83	82	85
27	92	85	85	81	82	82	81	85	85	88	79	82	94
28	85	88	86	83	90	81	84	84	81	85	80	82	86
29	87	84	89	80	80	81	86	83	80	83	76	84	91
30	83	83	84	76	80	78	82	85	83	87	78	83	85
31	88	83	84	80	81	78	83	87	83	88	75	80	91
32	88	83	85	83	83	82	83	84	81	89	78	82	90
33	87	83	85	84	80	81	83	85	80	86	81	78	82
34	87	84	82	86	80	81	81	82	85	84	78	82	82

35	81	83	85	81	80	83	79	82	80	83	85	78	79
----	----	----	----	----	----	----	----	----	----	----	----	----	----

- Sem 4

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	86	84	90	79	89	92	91	88	88	85	88	88	84
2	84	90	82	80	86	89	82	84	86	81	77	82	82
3	86	85	82	78	83	89	86	85	88	87	83	83	82
4	90	87	86	85	90	89	89	86	87	84	89	89	85
5	85	85	85	80	82	81	85	86	90	82	77	83	80
6	86	89	83	78	91	81	80	86	84	78	86	81	78
7	89	89	82	79	91	90	86	86	86	85	79	86	93
8	84	88	80	78	82	80	84	85	84	81	81	84	73
9	91	88	85	78	91	82	85	85	87	84	82	84	86
10	84	85	80	79	80	78	79	80	78	83	83	87	71
11	85	85	82	81	93	81	81	86	86	79	86	80	90
12	92	86	84	80	95	85	91	91	91	87	84	90	87
13	88	86	86	78	95	83	83	88	83	81	85	86	87
14	90	90	84	78	89	89	86	87	88	88	81	84	91
15	92	88	88	81	88	87	84	89	90	87	84	87	85
16	89	85	83	79	84	85	84	84	81	73	79	86	78
17	93	85	85	81	84	88	86	86	85	83	82	87	87
18	90	89	88	80	88	88	87	92	89	87	82	86	92
19	84	85	88	84	85	83	90	87	86	83	79	84	83
20	92	87	88	83	95	95	91	90	88	88	84	90	88
21	86	89	80	82	92	89	87	86	90	84	84	85	80
22	92	87	85	79	83	79	82	93	82	81	91	85	84
23	92	87	89	80	80	87	88	93	84	78	83	84	87
24	84	85	82	85	85	79	83	84	80	81	83	85	79
25	91	88	82	81	92	91	95	94	86	86	77	86	85
26	88	86	87	82	82	92	89	94	93	89	87	91	89
27	86	87	88	80	83	88	84	88	87	89	82	87	89
28	84	88	90	84	90	88	87	86	92	84	75	86	80
29	84	85	83	85	82	84	86	84	85	85	77	91	91
30	86	86	89	83	81	87	84	87	85	85	81	87	90
31	92	84	84	80	83	86	87	87	86	88	84	84	89
32	86	85	80	80	80	83	83	82	85	89	78	87	89
33	84	85	82	84	82	83	83	87	83	84	76	82	88



34	80	84	82	82	82	81	84	90	83	84	81	83	80
35	84	84	83	79	83	83	81	84	82	83	87	87	85

- Sem 5

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	88	86	85	84	90	90	88	87	83	88	83	92	87
2	78	89	84	85	81	80	85	87	81	86	81	90	91
3	91	87	88	83	81	85	83	90	84	89	80	90	80
4	92	87	87	84	95	88	82	90	84	88	79	90	91
5	78	87	85	83	81	80	81	86	86	86	81	83	80
6	78	89	84	81	83	80	80	85	80	78	80	80	78
7	92	88	87	83	87	82	80	87	84	88	80	86	82
8	78	88	85	81	84	80	80	83	81	83	80	87	84
9	93	88	85	84	86	86	83	89	84	88	78	86	90
10	78	85	82	80	82	80	80	81	80	86	80	81	42
11	78	85	83	83	81	80	81	85	80	83	84	92	95
12	94	88	95	84	91	86	89	90	89	87	81	92	95
13	78	88	85	82	87	84	85	85	82	85	80	84	91
14	92	90	88	82	86	86	85	85	84	90	80	88	90
15	84	88	90	85	86	85	85	87	84	78	80	80	82
16	78	87	84	80	80	80	80	84	80	88	80	81	78
17	89	86	88	83	86	83	83	91	82	87	82	87	88
18	88	89	87	83	91	84	87	92	86	88	82	85	87
19	87	86	89	89	86	83	86	85	82	86	78	87	84
20	93	88	91	87	88	88	85	85	85	90	80	88	90
21	81	88	87	88	84	83	83	85	84	90	84	86	89
22	87	88	87	81	81	80	81	88	83	88	81	85	84
23	89	86	85	83	81	82	83	85	81	87	80	84	90
24	86	87	85	84	82	80	83	85	83	88	82	86	91
25	88	88	93	85	93	92	86	95	84	89	80	88	88
26	90	87	93	82	86	90	86	91	86	89	81	88	83
27	94	88	91	83	89	85	82	87	85	90	85	87	89
28	91	87	89	81	89	90	83	91	84	89	86	86	86
29	78	86	86	83	81	86	87	86	85	89	80	86	90
30	89	87	85	83	87	82	84	86	83	89	80	86	79
31	89	85	88	81	92	85	85	85	84	89	80	89	90
32	94	86	86	83	90	88	84	87	86	90	81	89	95



33	80	85	86	85	82	82	83	88	84	87	80	86	89
34	79	86	83	82	84	80	80	86	85	87	76	84	78
35	78	85	85	82	81	80	80	87	80	86	84	86	80

## Kelas IPA 6

- Sem 3

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	84	84	87	88	84	81	80	85	85	83	83	86	92
2	87	85	87	85	80	83	87	90	85	91	78	85	92
3	86	83	86	88	80	85	79	85	83	83	79	82	79
4	83	83	83	86	85	79	70	85	83	83	80	81	79
5	88	83	88	88	82	83	80	84	83	78	81	81	78
6	89	83	87	88	85	90	83	85	82	84	83	86	87
7	92	83	84	84	84	86	84	86	83	91	79	88	88
8	84	83	85	84	80	80	79	85	78	85	75	84	79
9	89	83	88	81	80	80	81	83	78	84	76	88	92
10	90	83	87	82	80	80	80	87	78	80	75	85	96
11	91	83	82	80	82	86	90	86	80	82	77	84	93
12	81	83	88	86	82	94	90	86	79	86	76	81	82
13	86	83	81	89	83	80	79	84	85	86	82	84	79
14	86	83	84	82	81	81	79	84	79	81	81	85	90
15	93	83	86	85	81	74	82	86	83	89	76	85	83
16	91	88	84	78	87	80	85	87	84	90	81	84	89
17	90	83	85	85	80	78	80	89	83	86	78	86	81
18	83	83	83	80	80	78	83	84	78	90	77	85	91
19	89	83	83	82	10 0	85	87	82	82	87	77	81	87
20	86	83	87	87	80	81	79	83	78	79	78	86	85
21	81	83	84	90	81	83	83	83	82	85	84	84	78
22	85	83	85	90	80	80	78	82	81	83	77	81	92
23	81	83	81	90	80	78	70	83	81	83	81	82	78
24	86	83	80	78	80	78	85	84	78	83	75	78	85
25	87	83	87	80	80	81	80	82	78	89	76	81	92
26	85	83	86	79	80	83	80	85	80	85	78	80	79
27	85	83	81	88	84	84	86	82	82	88	77	80	85
28	87	83	82	90	80	86	80	83	88	86	79	81	87
29	86	85	87	81	80	85	79	83	80	85	75	82	89
30	86	84	89	77	80	78	83	84	78	82	75	89	79

31	88	83	88	86	80	81	80	82	78	87	75	86	90
32	87	85	87	84	91	85	84	86	84	90	79	88	89
33	85	83	81	78	91	79	81	82	81	89	78	87	90

- Sem 4

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	84	85	86	83	84	81	82	88	88	82	85	88	85
2	85	85	87	81	82	84	89	89	84	89	78	83	85
3	84	84	80	81	82	83	82	88	84	84	83	80	86
4	84	86	85	78	84	79	79	84	83	78	83	83	79
5	84	86	85	81	82	80	80	84	81	79	81	86	83
6	85	86	85	84	85	83	83	86	81	83	84	89	81
7	87	86	91	84	84	88	86	89	83	89	81	90	83
8	88	84	85	78	85	84	85	84	86	85	80	84	79
9	91	84	86	83	85	78	81	82	80	84	77	85	82
10	88	84	86	79	85	81	84	85	81	81	81	85	90
11	84	84	87	79	83	82	90	83	80	87	77	88	93
12	89	84	81	88	85	82	89	84	82	85	80	86	90
13	86	84	84	78	83	80	81	84	84	88	83	83	78
14	88	84	88	79	81	80	83	85	86	84	88	82	86
15	84	86	90	82	81	78	85	82	87	87	83	86	85
16	91	90	90	79	90	88	85	85	89	92	85	80	90
17	90	86	87	82	82	87	83	85	82	86	80	89	81
18	90	84	80	80	82	82	84	82	81	87	80	87	89
19	84	84	80	82	99	82	85	82	81	87	76	86	81
20	89	84	82	79	85	81	82	82	84	81	82	85	87
21	89	84	83	78	85	78	83	80	85	84	87	88	82
22	85	83	86	86	82	80	80	80	85	81	80	86	86
23	88	84	78	86	82	78	79	80	79	75	86	86	86
24	88	84	81	78	82	80	90	82	85	86	77	83	82
25	91	84	88	78	84	84	82	83	80	89	84	86	86
26	87	84	83	81	82	83	84	88	80	86	76	86	80
27	84	84	80	79	85	82	86	82	81	87	77	80	80
28	90	86	80	80	83	80	80	82	81	86	81	82	80
29	84	86	86	79	83	78	79	81	79	86	82	84	78
30	87	84	83	82	83	80	87	83	82	82	79	87	80
31	84	84	88	80	82	84	82	82	83	87	80	85	84



32	91	85	86	79	91	91	87	82	83	91	83	89	96
33	87	84	86	82	91	82	84	83	81	89	83	94	84

- Sem 5

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	88	87	89	84	80	80	87	86	82	86	82	80	89
2	89	86	84	88	86	84	83	90	80	87	80	89	92
3	78	86	82	82	80	80	80	83	80	82	84	80	80
4	81	86	82	84	81	80	81	85	84	85	80	80	81
5	80	86	88	85	80	84	82	84	83	88	85	80	80
6	88	86	87	84	88	92	83	88	84	87	83	90	88
7	90	87	88	88	87	84	84	88	85	89	85	83	92
8	92	87	84	86	91	89	82	88	82	87	80	87	82
9	88	85	87	87	81	83	84	85	80	87	81	86	85
10	92	85	87	82	81	82	82	88	80	88	80	84	91
11	90	86	86	83	81	84	86	88	83	87	82	86	93
12	83	85	84	88	86	88	86	90	80	89	83	82	84
13	84	85	85	86	81	85	82	83	81	87	85	85	81
14	83	88	83	85	82	84	80	84	81	89	83	86	81
15	89	88	90	88	84	85	84	83	83	89	80	86	85
16	94	88	84	84	81	84	85	89	85	88	82	88	93
17	88	87	85	87	87	86	84	85	80	90	82	84	81
18	89	85	90	82	86	84	83	86	81	90	82	86	86
19	89	85	87	87	88	87	84	85	80	86	81	85	87
20	82	86	84	82	83	80	82	80	81	85	80	80	80
21	78	86	84	82	81	84	80	82	82	81	84	80	84
22	84	86	83	88	80	83	82	85	80	86	78	84	87
23	78	86	80	80	83	80	80	82	78	79	81	80	80
24	87	85	84	83	84	82	83	84	82	87	80	86	86
25	84	85	86	87	85	82	84	84	81	89	83	89	86
26	88	85	84	87	86	82	84	87	80	88	82	87	83
27	89	85	86	85	92	82	81	86	81	87	82	91	88
28	89	87	88	83	88	82	85	80	80	87	83	91	92
29	84	86	85	84	81	82	81	84	84	88	82	82	90
30	89	86	86	85	85	88	83	89	84	87	80	83	88
31	89	85	85	83	84	85	84	90	82	87	80	85	89
32	89	85	89	83	81	88	82	87	82	87	81	88	94





33	91	88	86	86	86	82	83	82	84	89	80	87	90
----	----	----	----	----	----	----	----	----	----	----	----	----	----

Kelas IPA 7

- Sem 3

No	AGM	KWN	IND	ING	MAT	FIS	KIM	BIO	SJR	SNB	PJO	TIK	KB A
1	84	83	86	87	81	79	81	85	81	84	76	84	79
2	82	83	80	78	80	78	80	82	78	64	74	79	88
3	90	83	84	78	80	78	81	84	83	89	78	78	79
4	90	83	84	80	80	79	81	82	80	89	77	78	84
5	90	84	88	81	96	96	90	93	84	83	75	79	93
6	85	83	85	86	94	95	95	94	82	81	79	84	85
7	83	83	80	79	87	85	86	80	82	88	75	78	79
8	81	83	80	77	80	78	78	83	78	84	75	78	82
9	86	83	87	82	91	86	87	84	84	81	85	81	88
10	85	84	86	82	95	82	91	89	82	87	77	78	80
11	82	83	86	84	85	84	83	82	82	86	80	82	87
12	80	83	82	84	86	83	80	82	81	61	79	84	79
13	86	87	81	79	86	78	87	87	82	78	77	82	81
14	80	83	83	86	90	84	81	82	82	88	76	79	82
15	86	84	87	86	86	81	85	83	82	87	84	86	78
16	81	83	83	79	87	78	84	83	79	86	78	86	88
17	80	83	83	81	81	79	82	83	82	84	82	80	80
18	84	83	82	83	81	79	81	82	82	84	80	87	79
19	90	83	88	80	84	92	84	91	86	81	76	86	79
20	88	84	85	89	80	96	89	91	84	83	78	84	86
21	83	83	85	81	84	87	84	90	82	82	76	78	79
22	85	83	84	84	94	92	86	82	81	88	78	78	88
23	87	85	84	85	86	95	84	89	84	83	81	86	89
24	86	83	89	88	81	91	86	85	86	84	78	78	92
25	85	83	88	89	86	92	87	85	83	86	82	86	80
26	86	83	80	85	85	89	83	86	81	86	81	84	86
27	89	83	86	83	80	79	88	90	83	81	82	88	88
28	88	83	81	82	86	88	84	86	80	83	76	79	91
29	89	83	80	82	84	81	81	84	82	80	81	80	83
30	92	85	85	83	94	93	88	90	85	84	79	85	90
31	81	86	81	80	93	95	85	87	78	88	79	82	82
32	81	84	81	81	94	79	80	82	84	78	80	81	84



33	82	83	83	82	83	79	82	82	81	81	75	84	83
34	86	83	82	86	80	79	83	83	83	81	72	86	80
35	87	83	86	86	80	82	80	85	82	82	82	85	83

- Sem 4

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	84	86	85	88	81	79	83	85	83	87	78	88	87
2	86	84	80	79	80	78	80	82	80	78	77	81	83
3	89	86	86	81	81	78	84	85	81	90	79	87	78
4	92	84	81	80	81	78	81	82	80	88	75	80	84
5	94	87	86	80	96	93	92	89	80	85	77	87	89
6	91	85	85	87	94	82	94	92	81	85	81	89	80
7	84	84	85	80	89	88	84	82	80	88	84	84	78
8	84	84	80	79	81	78	78	81	79	83	75	80	84
9	90	86	84	81	92	82	88	84	81	86	81	81	86
10	87	84	84	83	95	89	92	85	82	87	81	82	80
11	92	85	81	85	85	86	86	83	84	87	76	87	83
12	84	84	82	81	88	80	83	83	80	67	81	87	78
13	93	88	84	84	88	82	88	85	81	78	85	83	81
14	84	84	80	78	91	88	79	82	87	87	81	80	82
15	89	86	84	81	86	88	79	80	80	88	83	90	83
16	85	84	84	81	88	88	86	82	83	88	82	84	84
17	84	85	88	79	83	82	80	82	82	87	78	80	78
18	84	84	80	78	83	87	81	88	88	78	81	80	78
19	84	85	85	82	84	92	87	91	92	84	79	81	80
20	84	86	89	82	82	86	90	90	92	86	83	83	81
21	84	84	80	78	84	88	82	82	79	86	76	80	78
22	89	86	83	85	94	85	86	84	85	88	80	83	84
23	90	88	92	88	86	91	84	83	87	83	79	88	82
24	84	86	90	87	82	90	85	82	82	84	77	80	80
25	87	85	88	89	86	90	87	87	80	86	81	84	83
26	88	85	82	88	85	91	85	88	80	86	84	80	81
27	86	84	86	86	84	90	89	87	87	83	87	88	85
28	86	84	80	78	86	84	88	87	88	85	79	88	83
29	87	85	78	79	85	82	80	82	86	86	80	82	78
30	92	88	88	83	94	91	89	88	88	86	78	89	99
31	84	86	82	84	93	88	84	86	80	89	76	81	87



32	84	84	80	85	94	92	79	81	81	81	81	88	84
33	84	84	80	79	83	79	82	82	79	81	80	82	83
34	86	84	84	90	82	79	80	82	79	84	84	83	83
35	87	85	85	82	82	80	80	84	79	85	85	82	83

- Sem 5

No	AGM	KWN	IND	ING	M AT	FI S	KI M	BI O	SJ R	SN B	PJ O	TI K	KB A
1	81	85	85	85	81	82	84	83	85	86	77	85	89
2	86	85	86	83	83	82	82	80	80	85	77	87	88
3	89,2	88	89	81	84	82	85	86	83	87	81	87	80
4	88	85	87	83	81	82	82	80	80	89	81	91	85
5	93,6	87	86	87	85	90	82	85	80	86	77	85	92
6	88,4	86	83	87	93	88	86	95	84	87	79	86	82
7	88	86	88	82	80	83	82	85	84	88	78	87	80
8	82	85	82	82	81	80	84	80	78	84	78	84	80
9	89	86	85	84	87	82	80	85	84	87	77	90	86
10	87	87	83	85	93	86	84	89	84	89	79	87	82
11	89	85	83	82	88	84	83	83	84	86	82	91	93
12	78	85	82	84	87	81	85	82	84	86	80	87	80
13	88	86	87	86	93	86	85	87	85	86	81	86	81
14	78	85	88	82	86	82	81	85	86	88	79	85	80
15	88	87	82	82	81	82	80	83	83	86	83	83	80
16	86	85	82	87	81	88	83	80	81	88	69	90	92
17	84	85	86	82	81	82	82	80	84	87	77	92	80
18	86	85	82	82	81	83	80	85	82	88	88	81	81
19	83	86	84	82	84	94	83	85	83	84	75	89	89
20	86	87	85	84	85	90	85	87	85	87	77	91	90
21	78	85	86	82	81	83	85	82	83	87	80	86	80
22	86	85	84	83	84	84	83	84	84	86	81	87	93
23	85	85	88	82	82	90	82	88	85	87	85	88	92
24	82	87	84	82	83	85	82	83	80	85	80	90	87
25	85	85	85	88	82	92	86	88	83	84	81	85	84
26	88	85	89	82	83	83	83	85	80	87	80	89	80
27	85	85	85	87	90	86	82	84	85	85	82	89	90
28	84	87	85	86	91	90	85	93	81	86	78	87	81
29	78	85	86	82	81	81	83	82	82	87	75	90	80
30	92	88	87	82	89	94	85	91	84	88	76	87	99



31	85	85	86	82	81	83	85	89	82	89	79	89	83
32	81	85	82	83	83	82	87	80	82	78	83	92	86
33	88	85	85	83	81	85	84	85	82	87	80	89	78
34	85	85	85	86	81	82	80	86	80	88	79	89	86
35	90	87	87	83	83	90	80	87	83	86	82	91	90



## LAMPIRAN 4 PEMETAAN KEMINATAN JURUSAN KULIAH

### Kelas Akselerasi

NO	PILIHAN 1 (JUR & PT)	PILIHAN 2 (JUR & PT)
1	ARSITEKTUR LANSEKAP (IPB) TEK. HASIL HUTAN (IPB)	KESMAS (UM)
2	TEK. SIPIL (UB) TEK. ARSITEKTUR (UB)	KEHUTANAN (UGM)
3	PEND. DOKTER (UB) GIZI (UB)	ILMU GIZI (UNAIR)
4	TEK. INDUSTRI (UB) FARMASI (UB)	PEND. KIMIA (UM)
5	PEND. DOKTER (UNDIP)	TEK. INDUSTRI (ITS)
6	GEOFISIKA (UB) TEK. KIMIA (UB)	BIOLOGI (UM)
7	STATISTIKA (UB)	MATEMATIKA (UM)
8	ILMU KEPERAWATAN (UB) PWK (UB)	BIOLOGI (UM)
9	TEK. IMFORMATIKA (UB) TEK. INDUSTRI (UB)	
10	TEK. INDUSTRI (UB) TEK. INFORMATIKA(UB)	TEK. INFORMATIKA (ITS)
11	TEK. ELEKTRO(UB) TEK. INFORMARTIKA (UB)	TEK. NUKLIR (UGM)
12	TEK. METALURGI & MATERIAL (ITS) BIOLOGI (ITS)	PEND. BIOLOGI (UM)
13	PEND. DOKTER (UGM)	DESAIN GRAFIS (ITS)
14	ILMU GIZI (IPB) TEK. PANGAN (IPB)	ILMU GIZI (UNAIR)

15	ILMU GIZI (UB) TEKNIK SIPIL (UB)	
16	STATISTIKA (UB) MATEMATIKA (UB)	MATEMATIKA (UM)
17	PEND. DOKTER (UNAIR)	PEND. DOKTER (UNS)

**Kelas IPA 1**

NO	PILIHAN 1 (JUR & PT)	PILIHAN 2 (JUR & PT)
1	STEI (ITB)	Informatika (ITS)
2	Pendidikan Dokter (UB) FKG (UB)	
3		
4		
5	Teknik Lingkungan (ITS) Teknik Geomatika (ITS)	Teknik Geodesi (UGM)
6	Sastra Inggris (UM) Pendidikan Bhs Ing (UM)	
7		
8	Teknik Industri (ITS) Sistem Informasi (ITS)	Teknik Industri (UGM)
9	FTTM (ITB)	Teknik Mesin (ITS)
10	FK (Unair) FKG (Unair)	
11	STEI (ITB)	
12	FK (UB) FKG (UB)	FK (Unair)
13	Teknik Sipil (UGM)	Perkapalan (ITS)
14	FTI (ITB)	
15	Arsitektur (UB)	Matematika Murni (UM)
16	FKG (UB) Gizi (UB)	

17		
18	FMIPA (ITB) STEI (ITB)	Teknik Fisika (ITS)
19	Teknik Mesin (ITS)	Teknik Mesin (UB) Pendidikan Dokter hewan (UB)
20	Pendidikan Dokter (UB)	
21	Kedokteran (UB) Gizi (UB)	Psikolog (UNS)
22	Teknik Sipil (ITS) PWK (ITS)	
23	FTTM (ITB) SAPPK (ITB)	Teknik Kimia (ITS)
24	FTI (ITB)	Teknik Industri (ITS)
25	Informatika (ITS)	
26	Gizi (UB) Statistika (UB)	
27	Teknik Mesin (ITS) Teknik Industri (ITS)	
28	FK Hewan (UB) Biologi (UB)	HI (Unair)
29	Psikolog (UM) DKV (UM)	
30	Arsitek Lansekap (ITS) Arsitek (ITS)	
31	Gizi (UB) Kebidanan (UB)	
32	Teknik Kimia (UGM) Teknik Pangan (UGM)	Manajemen Bisnis (ITS)
33	FK (UI)	FK (unair)
34	PWK (UB)	
35	FKG (UB)	

### Kelas IPA 2

NO	PILIHAN 1 (JUR & PT)	PILIHAN 2 (JUR & PT)
1	Sastra Jerman (UI)	Pend. Sastra Jerman (UM)

2	Matematika (UI)	Arsitektur (ITS)
3	Gizi (UB)	Bioteknologi (Unair)
4	Teknik Industri (UGM)	Teknik Industri (ITS)
5	Kedokteran gigi (UGM) Geofisika (UGM)	Geofisika (ITS)
6	Teknik Mesin (ITS)	Kedokteran (Unej)
7	Teknik Informatika (ITS) Sistem Informasi (ITS)	
8	Sistem Informasi (ITS) Teknik Informatika (ITS)	
9	Teknik Perkapalan (ITS) Kelautan (ITS)	
10	Mesin Elektro (ITS)	Teknik Mesin (UB)
11	Pendidikan Dokter (UB)	
12	FK (UB)	Teknik Perkapalan (ITS)
13	Kebidanan (UB) Gizi (UB)	Gizi (Unair)
14	STIH (ITB)	Biologi Murni (UM)
15	FTMD (ITB)	Teknik Mesin (ITS)
16	Pendidikan Dokter (UB)	
17	Teknik Fisika (ITS) Teknik Mesin (ITS)	
18	Teknik Industri (UB) Statistika (UB)	Pendidikan Matematika (UM)
19	Teknik Kimia (UGM) Farmasi (UGM)	Teknik Kimia (ITS)
20	Biologi Murni (UM)	Farmasi (unair)
21	Bisnis (UI)	Bisnis (UB)
22	Kedokteran (UB) Kebidanan (UB)	Kedokteran (Unej)
23	FK (UB) Psikolog (UB)	
24	Teknik Elektro (ITS) Teknik Mesin (ITS)	Teknik Elektro (UB)



25	Teknik Sipil (ITS)	Teknik Sipil (UGM)
26	Teknik Kimia (ITS) Teknik Industri (ITS)	Teknik Industri (UB)
27	FK (UGM) Gizi (UGM)	Gizi (Unair)
28	Desain Interior (ITS)	Arsitektur (UB)
29	Farmasi (UGM) Teknik Mesin (UGM)	Teknik Mesin (ITS)
30	Pendidikan Biologi (UM) Pendidikan IPA (UM)	
31	Agribisnis (IPB) Konservasi Sumber Daya (IPB)	Ilmu Keperawatan (Unair)
32	FK (UB) Teknik Informatika (UB)	FK (Unej)
33	Teknologi Bioproses (UI)	Peternakan (UB)
34	Teknik Sipil (ITB)	Teknik Sipil (ITS)

### Kelas IPA 3

NO	PILIHAN 1 (JUR & PT)	PILIHAN 2 (JUR & PT)
1	FK- UB	FK -UNEJ
2	FTTM - ITB	FTMD - ITB
3	TEKNIK FISIKA - UGM	TEKNIK GEOFISIKA - ITS
4	FK - UB	
5	FARMASI - UNAIR	TEKNIK INDUSTRI - ITS
6	STATISTIKA - IPB	STATISTIKA - ITS
7	FTMD - ITB	STATISTIKA - ITS
8	FARMASI - UGM	BIOLOGI - UGM
9	FKG - UNAIR	FARMASI - UGM
10	TEKNIK KIMIA - ITS	TEKNIK LINGKUNGAN - ITS

11	ADMINISTRASI PEMERINTAH - UB	MATEMATIKA - UM
12	KELAUTAN - ITB	BIOLOGI - UB
13	TEKNIK ELEKTRO -ITS	TEKNIK SIPIL- ITS
14	STEI - ITB	
15	HI -UNAIR	MANAJEMEN KEUANGAN - UNAIR
16	TEKNIK INDUSTRI - UGM	TEKNIK MESIN - UDAYANA
17	MANAJEMEN - IPB	AGRIBISNIS - UB
18	FK - UNAIR	FKG-UNAIR
19	TEKNIK NUKLIR - UGM	TEKNIK KIMIA - ITB
20	FK - UNEJ	FKH - UNAIR
21	FTTM - ITB	TEKNIK MESIN - ITS
22	AGRIBISNIS - UB	TEKNOLOGI INDUSTRI PERTANIAN - UB
23	TEKNIK KIMIA - UGM	TEKNIK ELEKTRO - ITS
24	TEKNIK ARSITEKTUR- UB	DKV - UB
25	FKG -UNAIR	FKG - UNEJ
26	FKG - UNAIR	
27	TEKNIK KIMIA - ITS	DESAIN INTERIOR - ITS
28	FK - UNEJ	KEBIDANAN - UNAIR
29	FK - UNAIR	
30	FTMD - ITB	TEKNIK MEKANTRONIKA - PENS
31	TEKNIK KIMIA - UB	TEKNIK KIMIA - POLTEK
32	SASTRA BAHASA INGGRIS - UM	PENDIDIKAN MATEMATIKA - UM
33	STEI - ITB	

34	TEKNOLOGI PANGAN - UB	GIZI - UB
35	FK- UNAIR	

#### Kelas IPA 4

NO	PILIHAN 1 (JUR & PT)	PILIHAN 2 (JUR & PT)
1	Teknik Kimia ( UB ) Ilmu & Teknologi ( UB ) pangan ( UB )	
2	FK ( UNS ) FKG ( UNS )	FK ( UB )
3	Teknik Perairan ( UB )	
4	Statistika ( ITS ) Teknik Kimia ( ITS )	Ilmu Gizi ( UNAIR )
5	Tidak Ikut SNMPTN	
6	FK (UNAIR) FKG (UNAIR)	FK (UNEJ)
7	Kesehatan Masyarakat (UNAIR) Farmasi (UNAIR)	Farmasi (UNAIR)
8	Psikolog (UNAIR)	
9	Teknik Industri (IPB) Pertanian (IPB)	FKG (UNEJ)
10	Gizi (IPB) Ilmu Pangan (IPB)	Gizi (UB)
11	Psikolog (UNPAD)	
12	Teknik Sipil (ITS)	Arsitektur (UB)
13		
14	Gizi (UB)	
15	Gizi (UB) Teknik Pangan (UB)	Sastra Jerman (UM)
16	FK (UNAIR)	Teknik Informatika (ITS)
17	FK (UNEJ)	
18	FKG (UNEJ)	

19	FK (UNSOED)	Arsitektur (ITS) Desain Interior (ITS)
20	Sekolah Bisnis dan Manajemen (ITB)	FSRD (ITB)
21	FK (UNS) Psikolog (UNS)	Psikolog (UNAIR)
22	SITH-S (ITB) FMIPA (ITB)	Kesehatan Masyarakat (UNAIR)
23	FK (UNUD) Fisioterapi (UNUD)	Gizi (UB)
24	Ilmu dan Teknologi Pangan (UB)	Teknik Mesin (UB) Biologi (UM)
25	Teknik Mesin (UM)	Manajemen (UM)
26	Arsitek (UB)	PWK (UB)
27	Arsitek (UB) Teknik Elektro (UB)	Teknik Mesin (UB)
28	FSRD (ITB)	Akun (UNAIR)
29	Kimia (UM) Pendidikan Kimia (UM)	Kimia (UIN)
30	Teknik Kelautan (ITS) Material Metalurgi (ITS)	FKH (UNAIR)
31	FKH (IPB) Arsitektur Lansekap (IPB)	Despro (ITS)
32	Agronomi dan Holtikultura (IPB)	Agroekotekno (UNEJ)
33	Farmasi (UNAIR)	Gizi (UB)
34	Teknik Industri (UB)	Teknik Mesin (UB)
35	FTTM ( ITB) FITB (ITB) FTSL (ITB)	

**Kelas IPA 5**

NO	PILIHAN 1 (JUR & PT)	PILIHAN 2 (JUR & PT)
1	TEKNIK ELEKTRO - UB	TEKNIK MESIN - UB
2	TEKNIK ELEKTRO - ITS	GEOFISIKA - ITS
3	ARSITEKTUR - ITS	DESAIN INTERIOR - ITS



4	ILMU GIZI - UB	STATISTIKA - ITS
5	ILMU GIZI - UB	KEHUTANAN - UGM
6	TEKNIK SISPIIL - UB	TEKNIK INDUSTRI - UB
7	PEND. DOKTER GIGI - UB	ILMU GIZI - UB
8	SBM - ITB	FTSL - ITB
9	STEI - ITB	SBM - ITB
10	ARSITEKTUR - ITS	DESAIN PRODUK - ITS
11	PEND. DOKTER - UNEJ	FARMASI - UNAIR
12	TEKNIK INDUSTRI - ITS	TEKNIK FISIKA - ITS
13	MANAJEMEN BISNIS - ITS	TEKNIK LINGKUNGAN - ITS
14	PEND. DOKTER GIGI - UNAIR	PEND. DOKTER GIGI - UNEJ
15	TEKNIK ELEKTRO - UB	TEKNIK MESIN - UB
16	TEKNIK SIPIL - ITS	TEKNIK INDUSTRI - ITS
17	TEKNIK INDUSTRI - UB	MANAJEMEN BISNIS - UB
18	PEND. DOKTER - UGM	TEKNIK INDUSTRI - ITS
19	PEND. DOKTER - UB	PEND. DOKTER - UNAIR
20	PWK - UGM	PWK - ITS
21	KEDOKTERAN HEWAN - IPB	KEHUTANAN - IPB
22	TEKNIK MESIN - UB	TEKNIK SIPIL - UB
23	KEHUTANAN - UGM	TEKNOLOGI INDUSTRI PERTANIAN - UGM
24	PEND. DOKTER - UNAIR	FARMASI - UNAIR
25	PEND. DOKTER - UNEJ	PEND. DOKTER GIGI - UNEJ
26	PEND. DOKTER - UNEJ	PEND. DOKTER - UDAYANA

27	TEKNIK FISIKA - ITS	TEKNIK ELEKTRO - UB
28	TEKNIK INDUSTRI - ITS	MANAJEMEN BISNIS - ITS
29	KEBIDANAN - UB	ILMU GIZI - UB
30	PEND. DOKTER - UB	ARSITEKTUR - UB
31	ARSITEKTUR - ITS	MATEMATIKA MURNI - UM
32	FTMD - ITB	TEKNIK INDUSTRI - ITS
33	PEND. DOKTER - UNAIR	ILMU GIZI - UNAIR
34	GEOFISIKA - ITS	MANAJEMEN BISNIS - ITS
35	PEND. DOKTER - UB	TEKNIK MESIN - UB

#### Kelas IPA 6

NO	PILIHAN 1 (JUR & PT)	PILIHAN 2 (JUR & PT)
1	TEK. INDUSTRI (ITS) TEK. FISIKA (ITS)	TEK. INDUSTRI (UB)
2	KEDOKTERAN (UB) GIZI (UB)	
3	TEK. INFORMATIKA (ITS) TEK. KELAUTAN (ITS)	
4	GEOLOGI (UGM) GEOFISIKA (UGM)	GEOFISIKA (UB)
5	TEK. SIPIL (UB)	
6	TEK. SIPIL (ITS)	
7	FARMASI (UNAIR) KIMIA (UNAIR)	TEK. KIMIA (ITS)
8	FARMASI (UNAIR) KIMIA (UNAIR)	KIMIA (UM)
9	TEK. INDUSTRI (ITS)	
10	KEDOKTERAN GIGI (UB) KEPERAWATAN (UB)	KEDOKTERAN GIGI (UNEJ)

11	BIOLOGI (UM)	FARMASI (UIN) BIOLOGI (UIN)
12	GEOFISIKA (ITS) PWK (ITS)	
13	FITB-ITB	TEK. PERKAPALAN (ITS)
14	FK (UNAIR)	FK (UNDIP)
15	KEDOKTERAN (UB)	
16	PENDIDIKAN BIOLOGI (UM) PENDIDIKAN MATEMATIKA (UM)	
17	GIZI KESEHATAN (UB) TEK. PANGAN (UB)	
18	FK (UB) GIZI (UB)	TEK. PANGAN (IPB)
19	TEK. KIMIA (ITS)	
20	TEK. METARULGI & MATERIAL (ITS) TEK. KELAUTAN (ITS)	
21	TEK. INDUSTRI (ITS) TEK. KOMPUTER (ITS)	
22	FK (UNDIP) ARSITEKTUR (UNDIP)	ARSITEKTUR (UNS)
23	STEI (ITB) FTSL(ITB)	TEK. INFORMATIKA (ITS)
24	KEDOKTERAN (UNEJ)	KEDOKTERAN (UNS)
25	ARSITEKTUR (UB) PWK (UB)	
26	DESIGN INTERIOR (ITS)	PSIKOLOGI (UB)
27	TEK. INFORMATIKA (ITS) SISTEM INFORMASI (ITS)	
28	AGRIBISNIS (UB)	
29	GEOFISIKA (ITS)	KEHUTANAN (UGM)
30	PWK (ITS)	EKONOMI (UNAIR)
31	ARSITEKTUR (ITS)	

32	TEK. KIMIA (ITS)	MATEMATIKA (UB)
33	ILMU KOMUNIKASI (UI) PSIKOLOGI (UI)	

### Kelas IPA 7

NO	PILIHAN 1 (JUR & PT)	PILIHAN 2 (JUR & PT)
1	ARSITEKTUR INTERIOR (UI) H. INTERNASIONAL (UI)	KEDOKTERAN (UNEJ)
2	BIOKIMIA (IPB)	TEK. INDUSTRI (ITS) TEK. LINGKUNGAN (ITS)
3	PWK (ITS)	TEK. SIPIL (UM)
4	FARMASI (UB) ILMU HUKUM (UB)	MATEMATIKA (UM)
5	MATEMATIKA (UGM)	MATEMATIKA (UB)
6	KEDOKTERAN (UNAIR)	KEDOKTERAN (UB)
7	PEND. MATEMATIKA (UM) PEND. FISIKA (UM)	
8	TEK. INDUSTRI (UB) GIZI (UB)	
9	TEK. INDUSTRI (ITS)	TEK. INDUSTRI (UB)
10	TEK. INFORMATIKA (UB) SISTEM INFORMASI (UB)	TEK. INFORMATIKA (UIN)
11	KEDOKTERAN (UB)	TEK. ELEKTRO (ITS) TEK. INFORMATIKA (ITS)
12	KEDOKTERAN HEWAN (UB)	KEDOKTERAN HEWAN (UNAIR)
13	KEDOKTERAN (UB)	
14	AGRIBISNIS (IPB) AGRONOMI (IPB)	AGRIBISNIS (UB)
15	TEK. PERKAPALAN (ITS) TEK. KELAUTAN (ITS)	TEK. ELEKTRO (UB)
16	FARMASI (UNAIR) AKUNTANSI (UNAIR)	FARMASI (UB)
17	GIZI (IPB)	KEPERAWATAN (UB)



18	MANAJEMEN KEHUTANAN (IPB) AGRONOMI (IPB)	
19	TEK. PANGAN (IPB)	TEK. LINGKUNGAN (ITS) TEK. MATERIAL (ITS)
20	MANAJEMEN BISNIS (UNAIR)	TEK. INDUSTRI (ITS)
21	KEDOKTERAN HEWAN (UB)	KEDOKTERAN (UNEJ)
22	AKUNTANSI (UNAIR) KESMAS (UNAIR)	KESMAS (UNEJ)
23	TEK. ELEKTRO (UB) TEK. INDUSTRI (UB)	
24	KEDOKTERAN (UB)	KEDOKTERAN (UNDIP)
25	KEDOKTERAN (UNS)	TEK. INDUSTRI (UB) GIZI (UB)
26	KEDOKTERAN (UNEJ)	KEDOKTERAN (UNUD)
27	SAINS (ITB)	BIOLOGI (UNAIR) KESMAS (UNAIR)
28	TEK. SIPIL (UB) FARMASI (UB)	KIMIA (UM)
29	ILMU KELAUTAN (UB) PSIKOLOGI (UB)	
30	MATEMATIKA (UB)	MATEMATIKA (UM)
31	TEKNIK ARSITEK (UGM) PWK (UGM)	TEKNIK ARSITEK (ITS)
32	TEK. ELEKTRO (ITS) DKV (ITS)	
33	KESMAS (UNDIP) TEK. GEOLOGI (UNDIP)	GIZI (UB)
34	AKUNTANSI (UNAIR) GIZI (UNAIR)	
35	PSIKOLOGI (UNPAD) SAS. JEPANG (UMPAD)	SAS. JEPANG (UNAIR)