

BAB IV

PERANCANGAN PERANGKAT LUNAK

Program aplikasi ini dibuat untuk mencari nilai eigen dan vektor eigen dari suatu matrik simetris. Dimana matrik ini didapat dari sistem persamaan homogen. Metode yang digunakan meliputi metode Fadjev-Leverrier, Bairstow, Power, dan Deflation, dimana penyelesaian dilakukan dengan kombinasi dua metode yaitu metode Fadjev-Leverrier - Bairstow dan metode Power-Deflation. Perbandingan masing-masing metode diukur berdasarkan waktu komputasi, analisis algoritma dan kompleksitas algoritma.

Langkah-langkah yang dilakukan dalam penelitian dengan menggunakan program aplikasi yang dibuat adalah sebagai berikut :

- a. Mula-mula dicari sistem persamaan homogen yang berukuran besar.
- b. Untuk input data matrik yang berukuran besar digunakan sistem pembangkitan bilangan secara random melalui komputer.
- c. Membuat penyelesaian sistem dengan metode Fadjev-Leverrier - Bairstow dan metode Power - Deflation.
- d. Sistem dicobakan untuk ukuran matrik yang bervariasi
- e. Hasil pengujian dianalisis waktu eksekusi, analisis algoritma dan kompleksitas algoritma.
- f. Disimpulkan metode mana yang terbaik.

4.1. Metode Perancangan

Program aplikasi harus mampu memberi fasilitas masukkan data yang representatif dan memberikan keluaran yang sesuai serta mempunyai tingkat kebenaran yang baik. Sehingga antara pengguna dan komputer dapat saling berinteraksi dan pengguna mampu merasakan kegunaan program tersebut.

Metode perancangan perangkat lunak yang digunakan untuk mengimplementasikan adalah dengan metode perancangan *top-down* yang mengacu kepada informasi yang akan ditampilkan. Perancangan tersebut menggunakan model *flowchart* untuk menggambarkan logika proses.

4.2. Hasil Perancangan

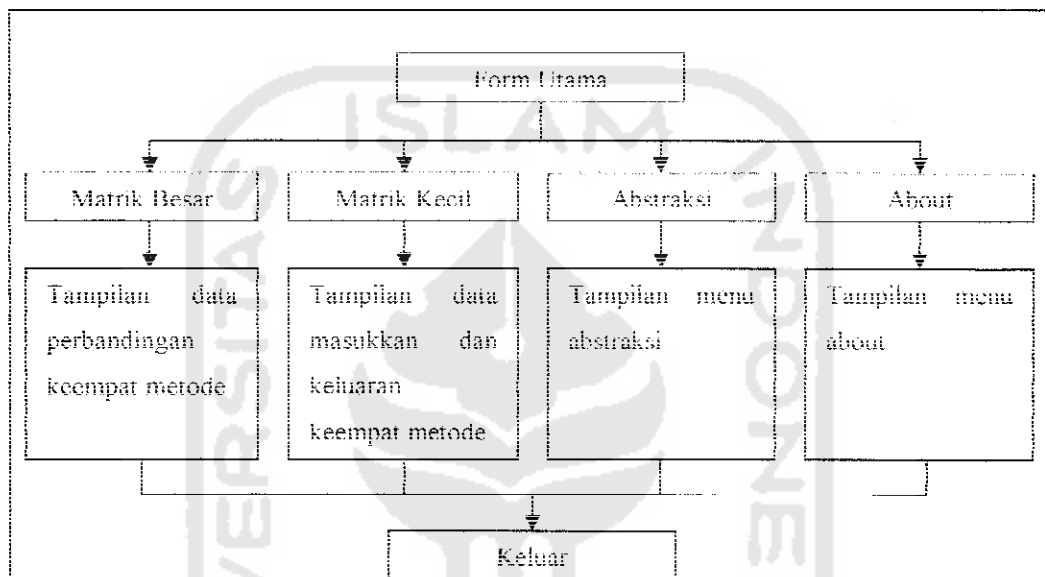
Sistem yang akan dibuat diharapkan dapat membantu pengguna dalam menyelesaikan persoalan untuk menghitung nilai eigen dan vektor eigen dari suatu matrik dengan hasil maksimal, yaitu lebih mendekati kebenaran. Juga menyediakan kemudahan pada saat pengguna mengoperasikan sistem tersebut.

4.2.1. Perancangan Ragam Dialog

Perancangan ragam dialog menggunakan perancangan berbasis visual yang memberikan kemudahan dalam hal pengontrolan format tampilan yang dapat dilakukan dengan mudah. Sehingga fleksibilitas tampilan dapat langsung dirasakan oleh perancang tampilan dan penggunanya.

Selain itu, perancangan ragam dialog menggunakan struktur pohon perintah. Metode tersebut lebih memudahkan pengguna dalam mengingat perintah dengan

mengelompokkan perintah-perintah tersebut menjadi satu kesatuan, dimana dalam waktu yang bersamaan akan dapat menyederhanakan proses pengambilan keputusan. Struktur pohon perancangan ragam dialog sesuai dengan gambar 4.1 berikut ini.



Gambar 4.1 Rancangan Ragam Dialog

4.2.2. Perancangan Tampilan Antarmuka

Pada bagian ini menjelaskan tentang tampilan-tampilan yang diinginkan dan akan diimplementasikan pada sistem tersebut. Dimana selanjutnya perancangan tampilan antarmuka ini digunakan sebagai acuan dalam membuat *interface* yang sesungguhnya.

4.2.2.1. Tampilan Menu Matrik Besar

Tampilan menu matrik besar berisi fasilitas untuk membandingkan keempat metode dari data yang diambil secara random. Pada menu ini, pengguna dapat menginputkan ukuran matrik yang diinginkan dan dapat melihat hasil dari perhitungan dari setiap metode yang diinginkan. Kemudian pengguna dapat melihat dengan jelas perbedaan dari hasil perhitungan untuk masing-masing metode berdasarkan waktu komputasi dan kompleksitas algoritma. Kompleksitas algoritma yang dimaksud adalah fungsi langkah yang terdiri dari banyaknya perulangan di dalam suatu algoritma dari setiap metode. Untuk lebih jelasnya akan diperlihatkan pada gambar 4.2 berikut ini.

Matrik .	Hasil .	
StringGrid	ListBox	ANALISIS PERBANDINGAN METODE 1. Fadeev-Leverrier dan Bairstow Waktu proses : <input type="text" value="Label 1"/> Kompleksitas : <input type="text"/>
		2. Power dan Deflation Waktu proses : <input type="text" value="Label 2"/> Kompleksitas : <input type="text"/>
Ukuran matrik .	<input type="text" value="ComboBox"/>	
<input type="button" value="Isi Matrik"/>	<input type="button" value="Cari"/>	

Gambar 4.2 Rancangan Tampilan Menu Matrik Besar

Pada rancangan tampilan menu matrik besar ini, berisi beberapa fasilitas untuk mengoperasikan menu tersebut, diantaranya :

1. *ComboBox* berfungsi untuk inialisasi ukuran dari matrik.
2. *Tombol Isi Matrik* berfungsi untuk mengisi matrik dengan suatu bilangan secara random.
3. *Tombol Cari* berfungsi untuk menjalankan sistem sesuai dengan masing-masing metode untuk mengetahui hasilnya.

4.2.2.2. Tampilan Menu Matrik Kecil

Tampilan menu matrik kecil berisi fasilitas yang hampir sama dengan menu matrik besar. Perbedaannya, pengguna dapat menginputkan sendiri data yang diinginkan untuk mengetahui hasil dari masing-masing metode. Tetapi ukuran matriknya dibatasi, yaitu hanya dari ordo 3 sampai ordo 10 saja.

Kemudian pengguna dapat melihat dengan jelas hasil perhitungan untuk masing-masing metode dengan tingkat ketelitian sampai tujuh dibelakang koma. Untuk lebih jelasnya akan diperlihatkan pada gambar 4.3 berikut ini

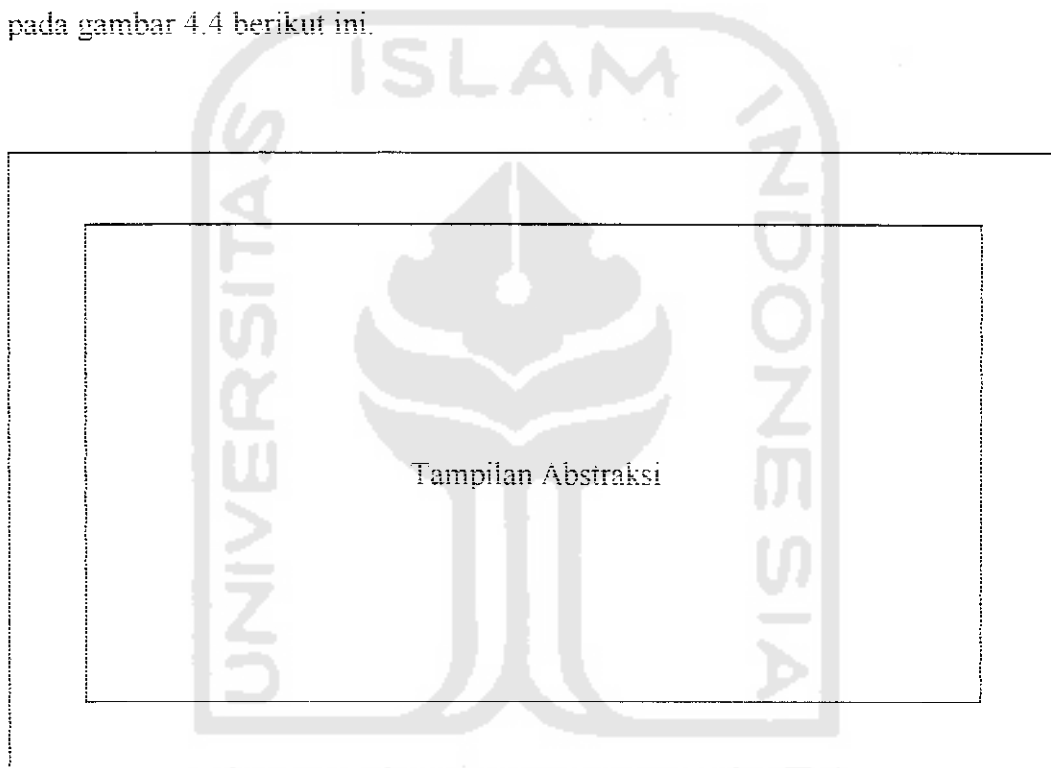
Gambar 4.3 Rancangan Tampilan Menu Matrik Kecil

Pada rancangan tampilan menu matrik kecil ini, berisi beberapa fasilitas untuk mengoperasikan menu tersebut, diantaranya :

1. *ComboBox* berfungsi untuk inialisasi ukuran dari matrik.
2. *Tombol Hapus* berfungsi untuk membersihkan data isian maupun data keluaran yang ada.
3. *Tombol Isi Matrik* berfungsi untuk mengisi matrik dengan suatu bilangan secara random.
4. *Tombol Cari* berfungsi untuk menjalankan sistem sesuai dengan masing-masing metode untuk mengetahui hasilnya.

4.2.2.3. Tampilan Menu Abstraksi

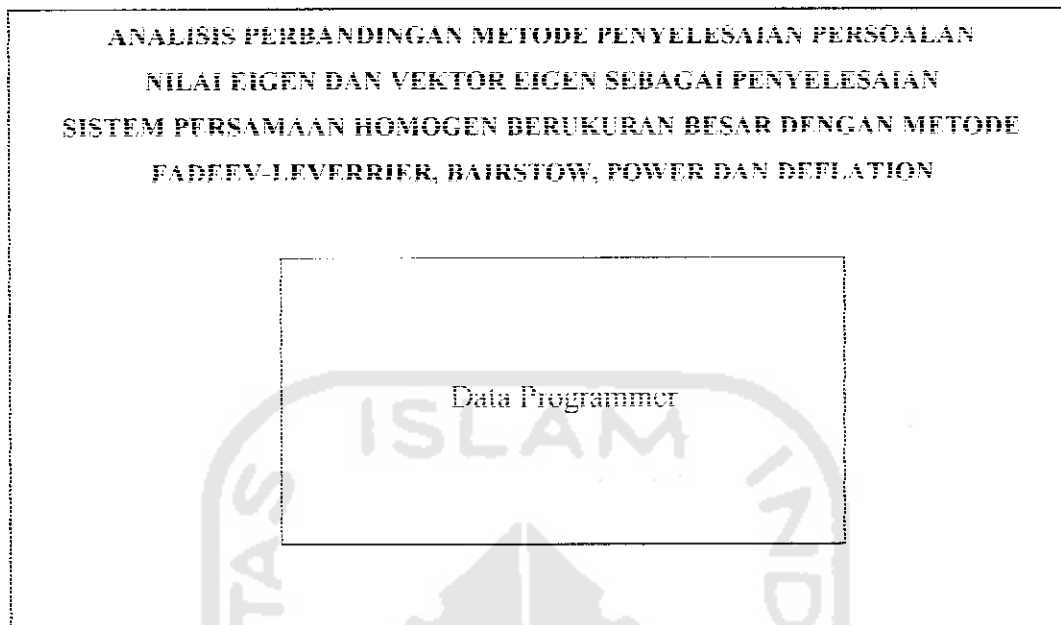
Tampilan menu abstraksi ini memberikan informasi dari abstraksi yang dibuat berdasarkan pembuatan tugas akhir ini. Dengan tujuan pengguna dapat mengetahui abstraksi atau gambaran singkat tentang mengapa sistem ini dibuat, sehingga pengguna dapat mengetahui manfaatnya. Rancangan tampilannya seperti pada gambar 4.4 berikut ini.



Gambar 4.4 Rancangan Tampilan Menu Abstraksi

4.2.2.4. Tampilan Menu Info

Tampilan menu info ini memberikan informasi tentang pembuat sistem analisis perbandingan persoalan nilai eigen dan vektor eigen sebagai penyelesaian sistem persamaan homogen berukuran besar. Rancangan tampilannya seperti pada gambar 4.5 berikut ini.



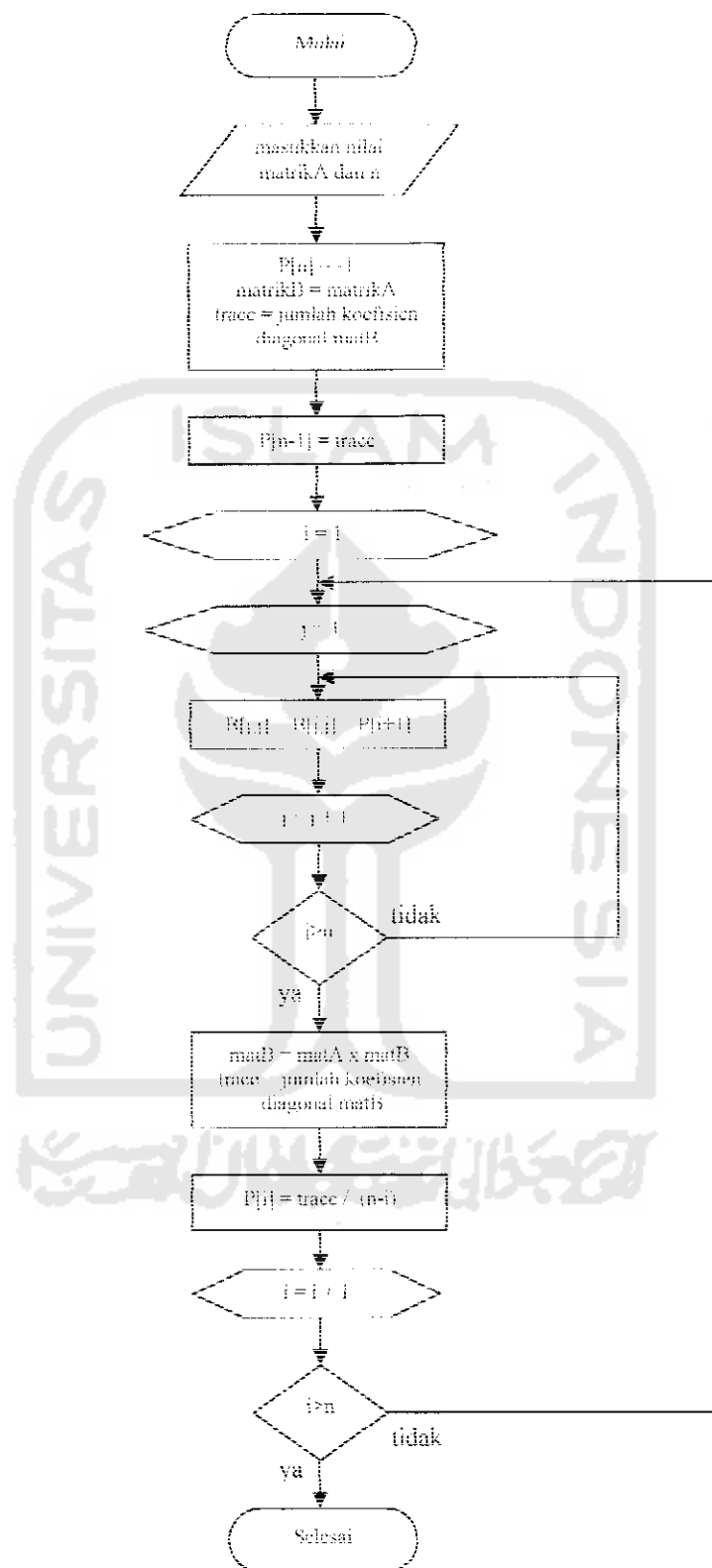
(Gambar 4.5 Rancangan Tampilan Menu Abstraksi)

4.3 Diagram Alir (*flow chart*)

Diagram alir (*flow chart*) merupakan bagan aliran sistem perhitungan nilai eigen dan vektor eigen dengan simbol tertentu yang menggambarkan urutan-urutan kerja prosedur atau fungsi yang ada dalam sistem. Pada tahap perancangan sistem, penggunaan simbol membantu komunikasi antara pengguna dengan sistem, sehingga diharapkan pengguna dapat memahami sistem secara logika. dengan alat desain *flow chart* memungkinkan penggambaran sistem secara keseluruhan, mulai awal sistem, memasukkan data, hingga keluaran sistem.

4.3.1. Metode Faddeev-Leverrier

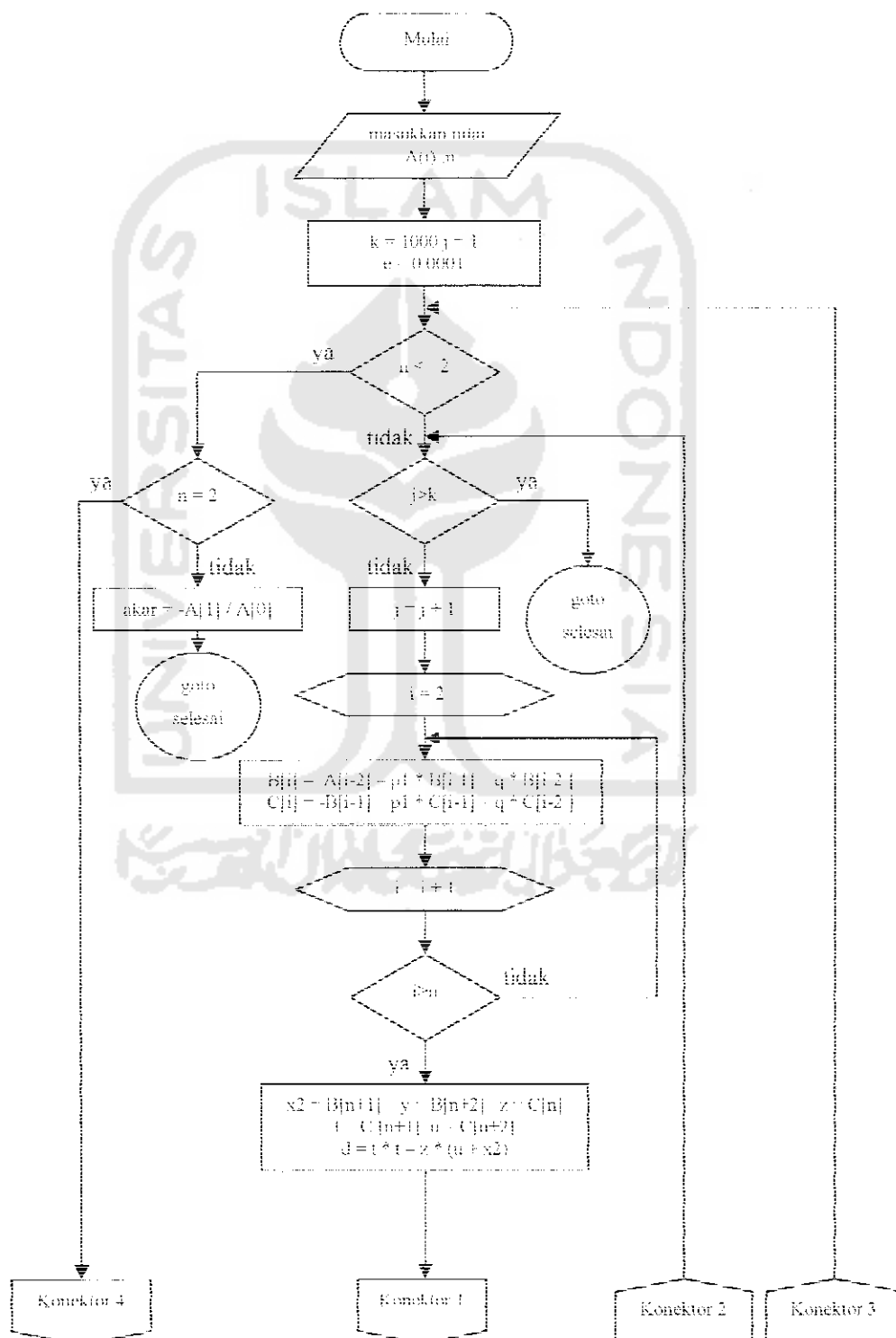
Diagram alir dari metode Faddeev-Leverrier dapat dilihat pada gambar 4.6 berikut :

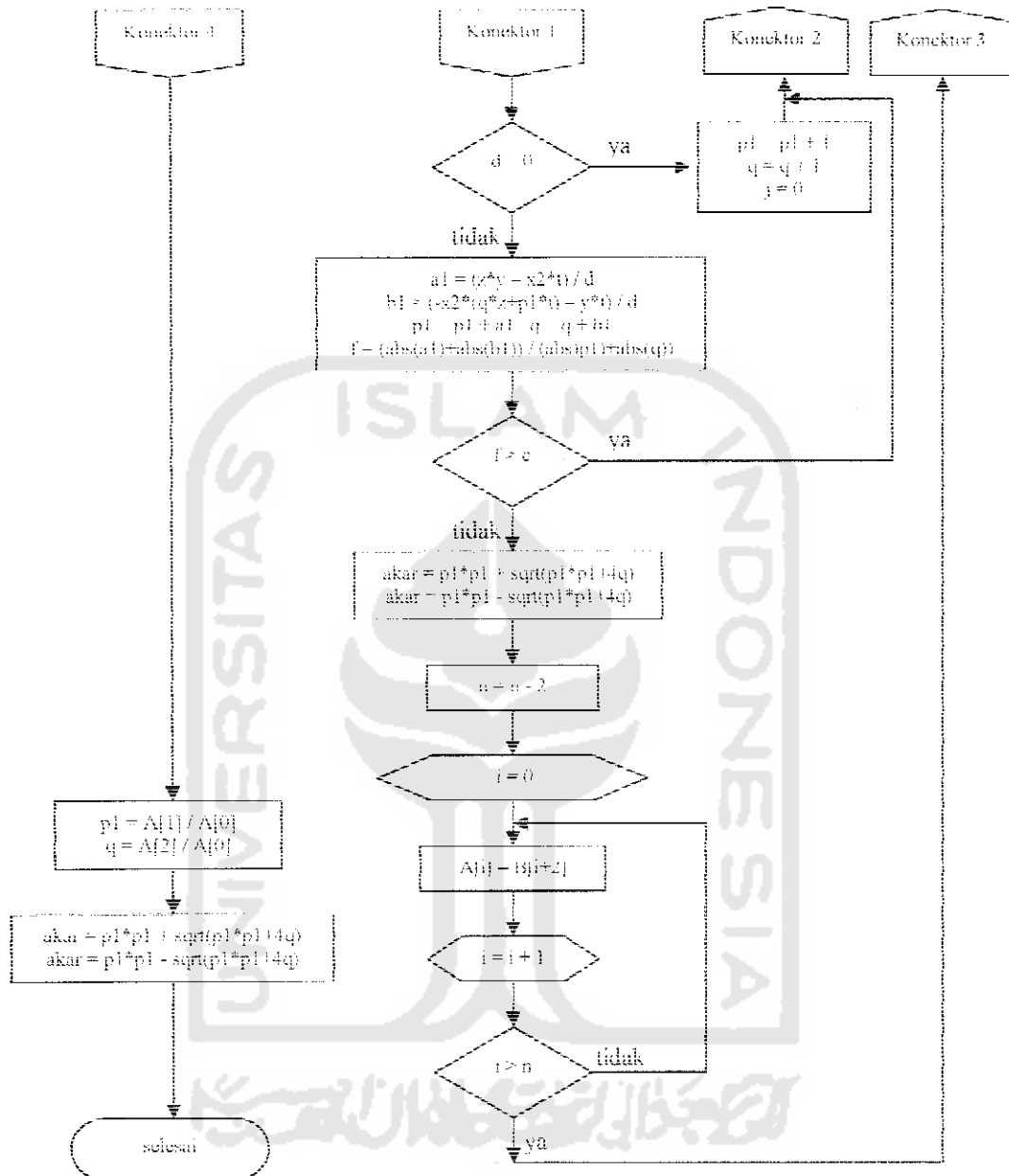


Gambar 4.6 Diagram Alir Metode Faddeev-Leverrier

4.3.2. Metode Bairstow

Diagram alir proses dari metode Bairstow dapat dilihat pada gambar 4.7 berikut :



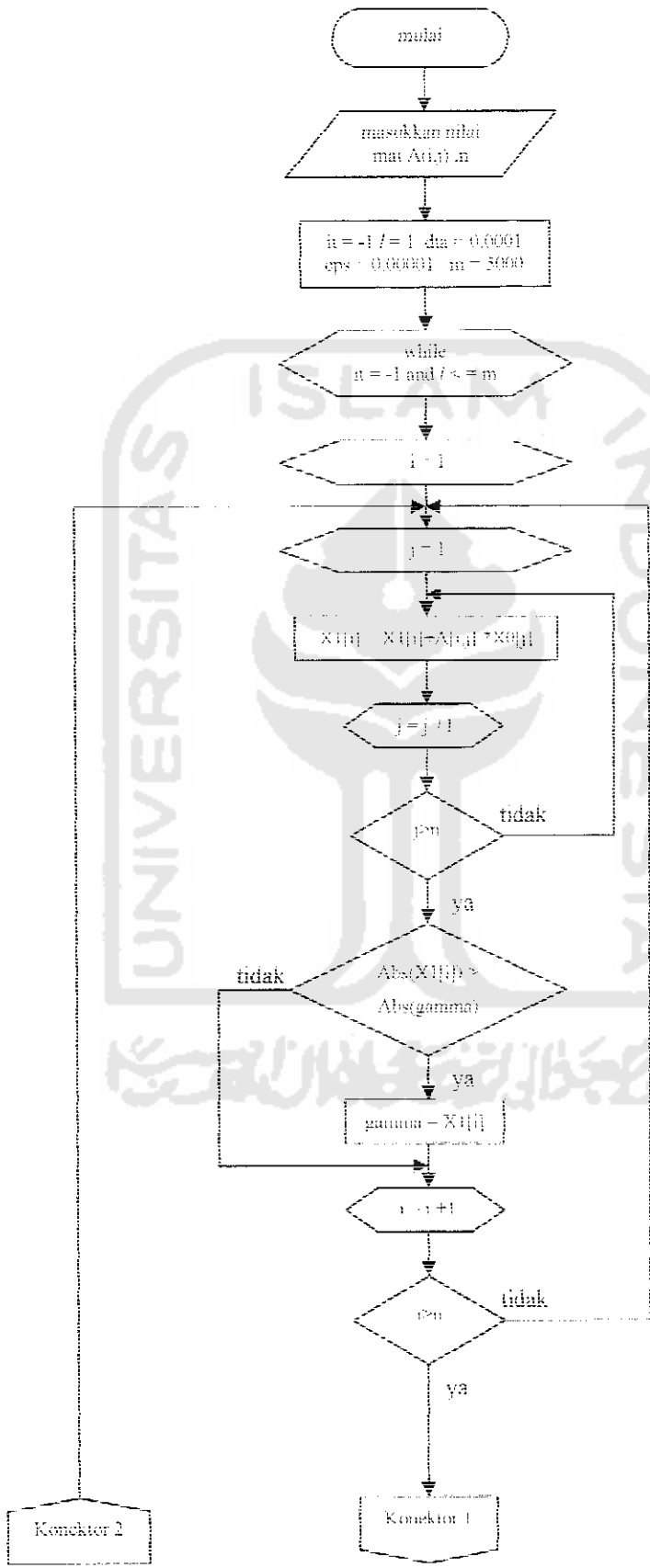


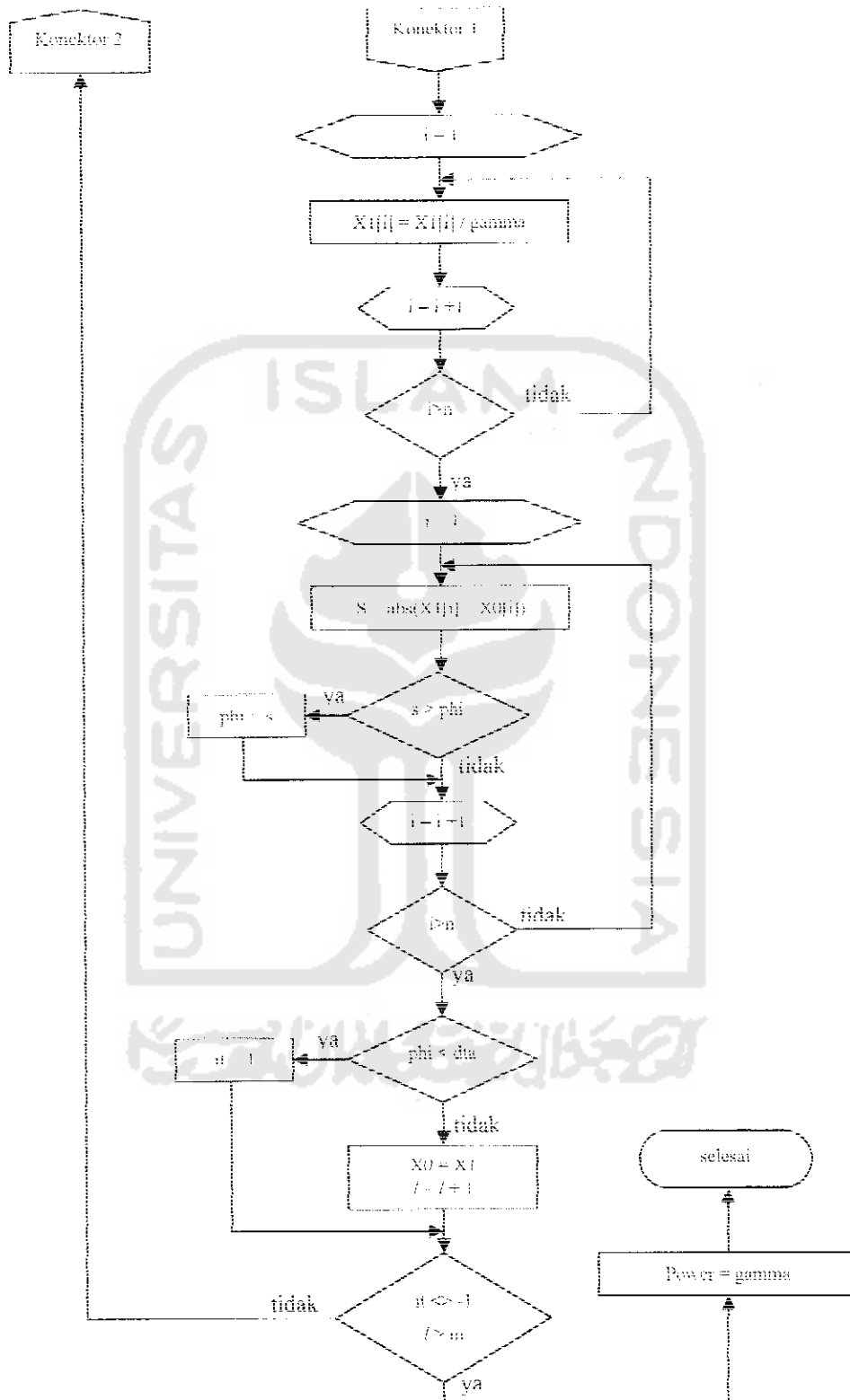
Gambar 4.7 Diagram Alir Metode Bairstow

4.3.3. Metode Power

Diagram alir dari metode Power dapat dilihat pada gambar 4.8 berikut :

MILIK
PERPUSTAKAAN-FTI-UII
YOGYAKARTA

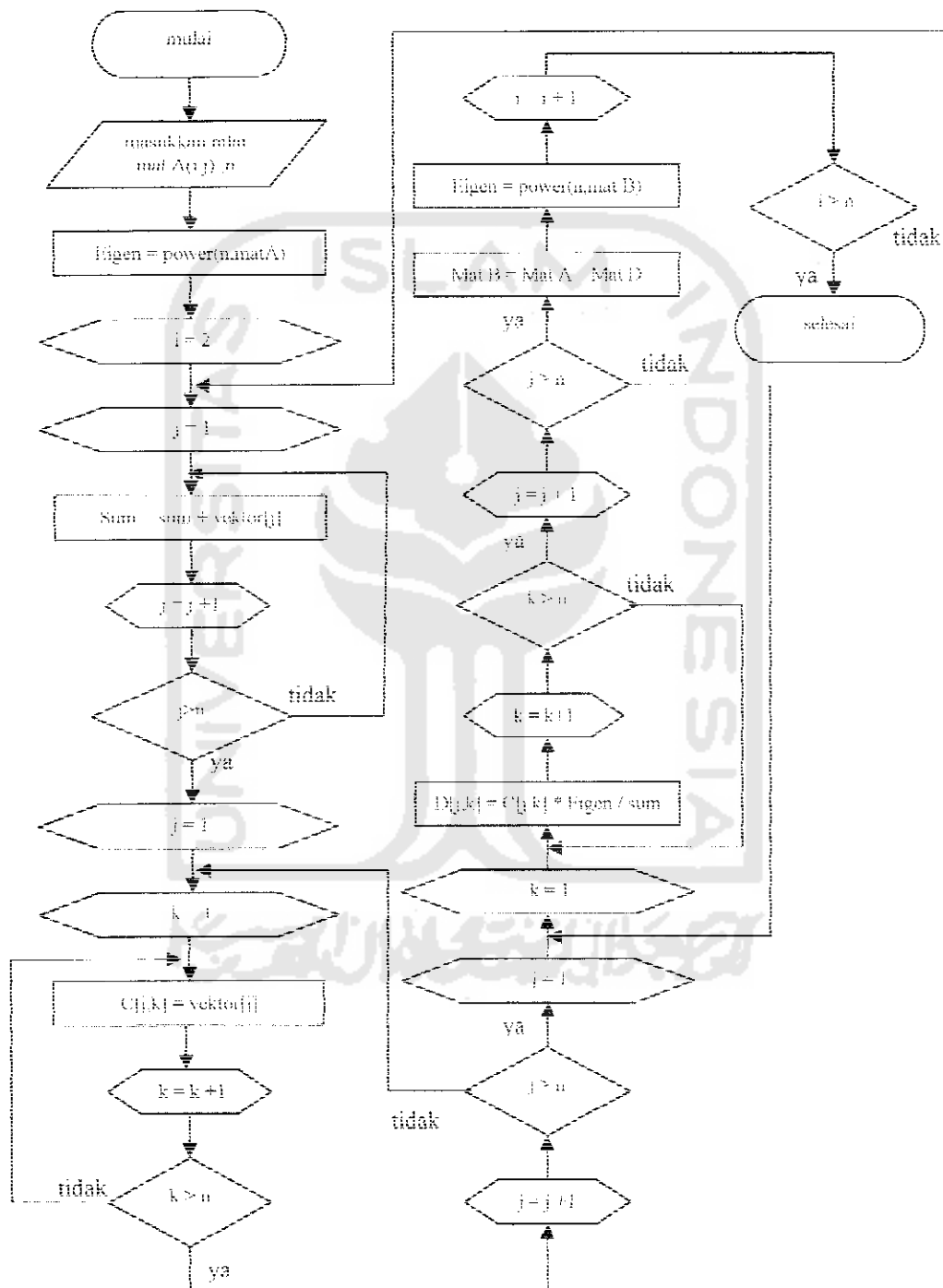




Gambar 1.8 Diagram Alir Metode Power

4.3.4. Metode Deflation

Diagram alir dari metode Deflation dapat dilihat pada gambar 4.9 berikut :



Gambar 4.9 Diagram Alir Metode Deflation