

## **BAB III**

### **PROSES PEMROGRAMAN**

#### **3.1. Umum**

Di masa globalisasi seperti sekarang ini, program komputer merupakan sarana yang tepat untuk membantu menyelesaikan perhitungan agar lebih cepat dan teliti. Program itu berisi langkah-langkah yang harus dilalui untuk menyelesaikan berbagai persoalan, baik perhitungan matematika maupun pengolahan data.

Untuk mempermudah proses pembuatan program, terlebih dahulu disusun langkah-langkah penyelesaian yang akan dikerjakan. Langkah-langkah tersebut ditransfer ke dalam bentuk *flow chart*, yang akan mempermudah untuk diterjemahkan ke dalam bahasa program. Dalam tugas akhir ini bahasa program yang digunakan adalah Turbo Basic.

#### **3.2. Langkah-langkah Pemrograman Analisa Struktur (Matriks)**

##### **3.2.1. Membaca dan Menyimpan Data Struktur**

Input data yang diperlukan dan harus dibaca adalah: NE (jumlah elemen), NJ (jumlah joint) dan NLC (jumlah kondisi pembebanan). Kemudian data struktur, data dukungan dan data beban, dibaca dan disimpan untuk diproses.

Data struktur meliputi informasi batang yaitu MINC(1,1) (joint pada ujung awal batang yang ditinjau), MINC(2,1) (joint pada ujung akhir batang) dan material batang yaitu, AREA(luas batang), ZI(inersia), EMOD(modulus elastis batang), ID ( jenis batang), ELENG (panjang batang), S (faktor kekakuan batang).

Data dukungan meliputi JNUM (nomor joint yang menjadi dukungan) dan JDIR (arah vektor gaya yang bebas). Untuk data beban batang meliputi batang yang terbebani, jenis beban, arah beban, jarak dan besar beban. Sedangkan data beban joint meliputi arah beban dan besar beban.

### 3.2.2 Mencari Jumlah Persamaan Simultan

Semua JCODE (kode matriks joint) dianggap mempunyai vektor deformasi, artinya dianggap tidak terkekang (bernilai 1). Kemudian membaca input dukungan, jika input dukungan memberikan nilai maka JCODE-nya diberi nilai nol. Artinya pada JCODE terjadi pengekangan. Semua data JCODE dibaca lagi, jika nilai JCODE tidak sama dengan nol maka mempunyai nilai  $NEQ = NEQ + 1$ , dengan NEQ mula-mula bernilai nol. Pada joint yang ada elemen Springnya, nilai NEQ arah x dan arah y nilainya tetap dengan NEQ yang sebelumnya dengan membaca data JCODE yang berada pada koordinat yang sama, tetapi nilai NEQ arah perputaran tetap ditambah 1. Misalnya untuk JCODE (2) = [1 2 3], maka nilai JCODE (3) = [1 2 4].

### 3.2.3 Menentukan Hal Bandwidth (MBD)

Semua kode matriks batang harus ditentukan lebih dulu, dengan cara membaca batang tersebut sama seperti nilai JCODE-nya.

$$\text{MCODE}(L,1) = \text{JCODE}(L,\text{MINC}(1,I))$$

$$\text{MCODE}(L+3,1) = \text{JCODE}(L,\text{MINC}(2,I))$$

dengan :  $I = 1, \text{NE}$  dan  $L = 1,3$

Dari nilai setiap nilai MCODE dalam satu lajur kolom dibanding dengan MCODE lainnya dalam satu kolom. Untuk memperoleh MBD digunakan ketentuan sebagai berikut ini.

- a. Jika  $\text{MCODE}(J,I) \neq 0$  dan  $\text{MCODE}(K,I) \neq 0$ , maka digunakan :

$$\text{MBD} = \text{ABS}(\text{Mcode}(J,I) - \text{MCODE}(K,J))$$

dengan,  $I = 1, \text{NE}$ ;  $J = 1,6$ ; dan  $K = J+1,6$

- b. Jika MCODE keduanya = 0, maka nilai  $\text{MBD} = 0$

Nilai MBD yang dipakai adalah nilai MBD yang terbesar.

### 3.2.4 Menghitung Matriks Transformasi Batang

Matriks transformasi diperoleh dari input data koordinat joint yaitu :

- $X(1, \text{Nomor Joint})$  sebagai koordinat sumbu x.
- $X(2, \text{Nomor Joint})$  sebagai koordinat sumbu y.

Persamaannya adalah :

$$C1(I) = \frac{X(1,K) - X(1,J)}{\text{ELENG}(I)}$$

$$C2(I) = \frac{X(2,K) - X(2,J)}{\text{ELENG}(I)}$$

dengan :

$$\text{ELENG}(I) = \text{panjang batang} = \sqrt{\{X(1,K) - X(1,J)\}^2 + \{X(2,K) - X(2,J)\}^2}$$

$$C1 = \cos \theta = c; \quad C2 = \sin \theta = s$$

$$J = \text{MINC}(1, I); \quad K = \text{MINC}(2, I)$$

### 3.2.5 Pembebanan

Pembebanan dibedakan menjadi dua macam, yaitu pembebanan joint dan pembebanan batang. Untuk pembebanan joint dapat diperoleh dengan langkah sebagai berikut ini.

Input data yang dibaca komputer adalah JNUM (nomer joint), JDIR (arah beban yang bekerja) dan FORCE (besar beban, bernilai positif jika kekanan atau keatas dan negatif bila kekiri atau kebawah). FORCE ini merupakan beban luar yang akan berpengaruh terhadap nilai deformasi dan dinotasikan sebagai Q(K).

$$Q(K) = \text{FORCE}, \quad \text{dimana } K = \text{JCODE}(JDIR, JNUM)$$

Sedangkan untuk beban batang, input data yang dibaca adalah MN (nomor batang), MAT (Tipe pembebanan, 1 untuk beban terpusat dan 2 untuk beban merata), DIST (jarak dari joint batang awal) dan ACT (besar beban). Untuk menghitung gaya jepit ujung akibat beban luar digunakan persamaan (2.53) dan (2.54) yaitu :

a. Beban titik.

$$F(1, MN) = F(1, MN) \text{ sebelumnya}$$

$$F(2, MN) = F(2, MN) \text{ sebelumnya} + \text{ACT} (-1 - A^2(2A - 3))$$

$$F(3, MN) = F(3, MN) \text{ sebelumnya} + \text{ACT} (-\text{DIST} (1 - A)^2)$$

$$F(4, MN) = F(4, MN) \text{ sebelumnya}$$

$$F(5, MN) = F(5, MN) \text{ sebelumnya} + \text{ACT} (A^2 (2A - 3))$$

$$F(6, MN) = F(6, MN) \text{ sebelumnya} + \text{ACT} (\text{ELENG. } A^2 (1 - A))$$

b. Beban merata.

$$F(1,MN) = F(1,MN) \text{ sebelumnya}$$

$$F(2,MN) = F(2,MN) \text{ sebelumnya} - \frac{1}{2} \cdot \text{ACT. ELENG}$$

$$F(3,MN) = F(3,MN) \text{ sebelumnya} - \frac{1}{12} \cdot \text{ACT. ELENG}$$

$$F(4,MN) = F(4,MN) \text{ sebelumnya}$$

$$F(5,MN) = F(5,MN) \text{ sebelumnya} - \frac{1}{2} \cdot \text{ACT. ELENG}$$

$$F(6,MN) = F(6,MN) \text{ sebelumnya} - \frac{1}{12} \cdot \text{ACT. ELENG}$$

### 3.2.6 Menyusun matriks beban luar total

Matriks beban luar total merupakan penggabungan antara beban joint dan beban batang kemudian ditransformasikan dengan memanfaatkan statemen function pada bahasa Turbo BASIC.

$$Q(\text{MCODE}(1,I)) = Q(\text{MCODE}(1,I)) \text{ sebelumnya} - (C1(I) \cdot F(1,I) - C2(I) \cdot F(2,I))$$

$$Q(\text{MCODE}(2,I)) = Q(\text{MCODE}(2,I)) \text{ sebelumnya} - (C1(I) \cdot F(1,I) - C2(I) \cdot F(2,I))$$

$$Q(\text{MCODE}(3,I)) = Q(\text{MCODE}(3,I)) \text{ sebelumnya} - F(3,I)$$

$$Q(\text{MCODE}(4,I)) = Q(\text{MCODE}(4,I)) \text{ sebelumnya} - (C1(I) \cdot F(4,I) - C2(I) \cdot F(5,I))$$

$$Q(\text{MCODE}(5,I)) = Q(\text{MCODE}(5,I)) \text{ sebelumnya} - (C1(I) \cdot F(4,I) - C2(I) \cdot F(5,I))$$

$$Q(\text{MCODE}(6,I)) = Q(\text{MCODE}(6,I)) \text{ sebelumnya} - F(6,I)$$

dengan : I = nomor batang

### 3.2.7 Membentuk matriks kekakuan sistem struktur

Untuk menghitung matriks kekakuan menggunakan persamaan (2.50) dengan memanfaatkan statement index matriks pada bahasa komputer, sehingga

akan dapat diperoleh matriks kekakuan batang/elemen pada sistem koordinat global. Matriks kekakuan batang dibagi menjadi dua batang/elemen, yaitu :

### 1. Matriks kekakuan batang biasa

$$\text{INDEX} = \begin{bmatrix} 1 & 2 & 4 & -1 & -2 & 4 \\ & 3 & 5 & -2 & -3 & 5 \\ & & 6 & -4 & -5 & 7 \\ & & & 1 & 2 & -4 \\ & & & & 3 & -5 \\ \text{sim} & & & & & 6 \end{bmatrix}$$

dimana :

$$1 = g1 = \alpha(\beta c^2 + 12C2^2)$$

$$2 = g2 = \alpha C1.C2(\beta - 12)$$

$$3 = g3 = \alpha(\beta C2^2 + 12.C1^2)$$

$$4 = g4 = -\alpha 6 \text{ ELENG } C2$$

$$5 = g5 = \alpha 6 \text{ ELENG } C1$$

$$6 = g6 = \alpha 4 \text{ ELENG}^2$$

$$7 = g7 = \alpha 2 \text{ ELENG}^2$$

### 2. Matriks kekakuan batang Spring

$$\text{INDEX} = \begin{bmatrix} 1 & 2 & 4 & -1 & -2 & 4 \\ & 3 & 5 & -2 & -3 & 5 \\ & & 6 & -4 & -5 & 7 \\ & & & 1 & 2 & -4 \\ & & & & 3 & -5 \\ \text{sim} & & & & & 6 \end{bmatrix}$$

dengan :

$$1 = g1 = 0$$

$$2 = g2 = 0$$

$$3 = g3 = 0$$

$$4 = g4 = 0$$

$$5 = g5 = 0$$

$$6 = g6 = 4 \cdot \frac{EMOD.ZI}{ELENG} \cdot S$$

$$7 = g7 = -4 \cdot \frac{EMOD.ZI}{ELENG} \cdot S$$

dengan :  $\alpha = \frac{EMOD.ZI}{ELENG^2}$ ,  $\beta = \frac{AREA.ELENG^2}{ZI}$ , S=Faktor kekakuan Spring

Jika  $J = MCODE(JE, \text{nomor batang}) \neq 0$ ,  $JE = I,6$ ; dan  $I = MCODE(IE, \text{nomor batang}) \neq 0$ ,  $IE = JE.6$ ; maka dapat dicari nilai  $g$  pada matriks dengan index yang telah ditentukan di atas. Untuk mendapatkan nilai index seperti di atas yaitu ada yang bernilai negatif sedang rumus  $g$  yang digunakan positif, maka digunakan statement float untuk menjadikan real suatu nilai. Letak matriks kekakuannya dapat dicari dengan persamaan :

$$SS(J,K) = SS(J,K) \text{sebelumnya} + g(\text{AbsINDEX}(IE,JE)) \text{FLOAT} \frac{\text{INDEX}(JE,IE)}{\text{AbsINDEX}(IE,JE)}$$

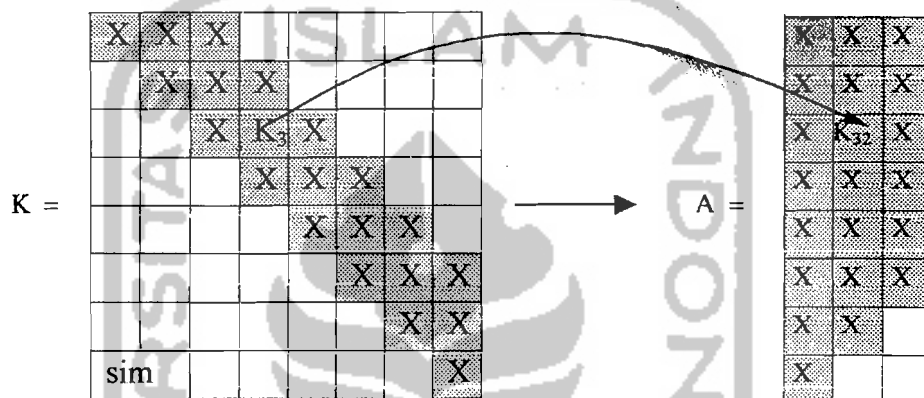
Nilai  $k$  adalah letak kolom yang disimpan dalam band matriks. Nilai  $k$  berfungsi untuk menyederhanakan bentuk matrik sehingga dapat menghemat waktu komputer dan penyimpanan (memory). Bentuk penyederhanaan tersebut disebut jalur atas (*upper band*) matriks kekakuan keseluruhan. Matriks  $n \times n$  dapat

dijadikan suatu array segi empat  $n \times b$ . Nilai  $b$  adalah setengah lebar jalur (*half band width*). Gambar 3.1 menunjukkan proses penyimpanan kedalam suatu matriks segi empat.

Hubungan antara elemen  $K$  dan  $A$  adalah

$$K_{ij} = A_{ik}, k = j - i + 1$$

Baris (i) untuk  $k$  sama dengan baris (I) pada  $A$



Gambar 3.1. Proses penyimpanan kedalam baris bandwidth matriks.

### 3.2.8 Menyelesaikan persamaan matriks

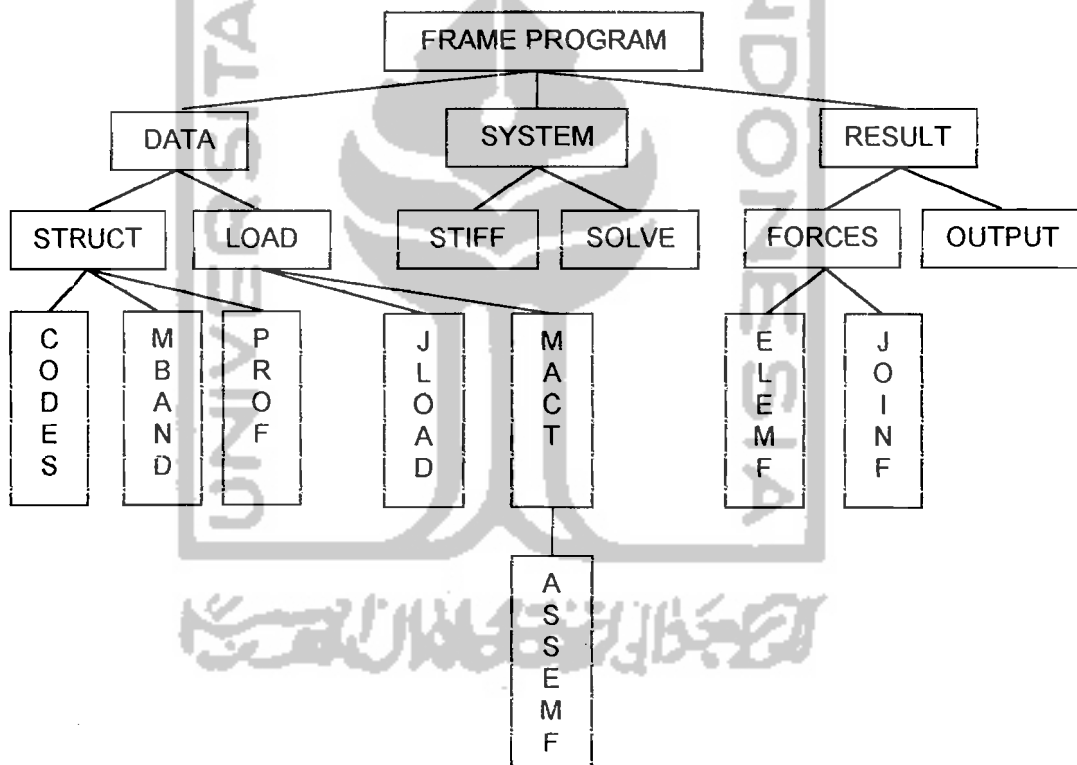
Persamaan matriks diselesaikan dengan menggunakan metode eliminasi Gauss. Vektor  $Q$  yang semula sebagai vektor matriks beban luar, keluar sebagai vektor matriks displesmen. Kemudian matriks displesmen masing-masing batang disusun yang diambil dari matrik displesmen struktur dengan bantuan MCODE masing-masing batang. Kemudian dengan persamaan  $d = A.D$  didapat displesment masing-masing batang. Nilai  $d$  dari persamaan itu dipakai untuk menghitung gaya batang dengan persamaan  $f = k \cdot d$ .



### 3.3. Flow Chart

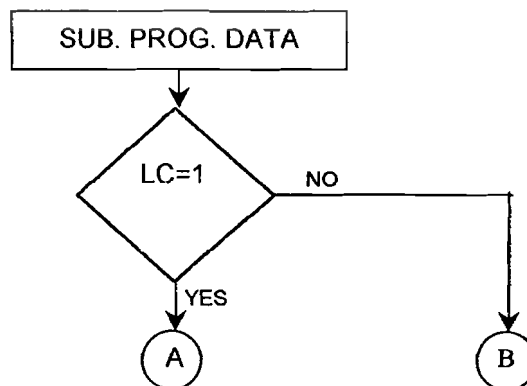
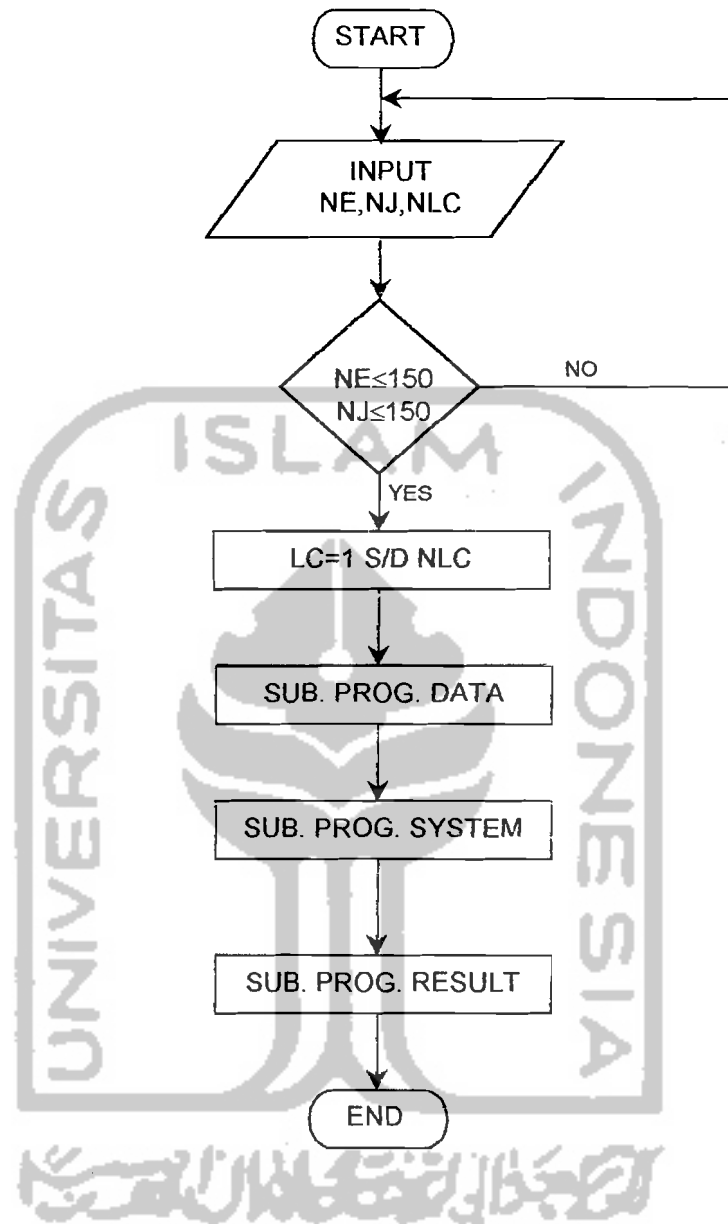
#### 3.3.1. Struktur Program

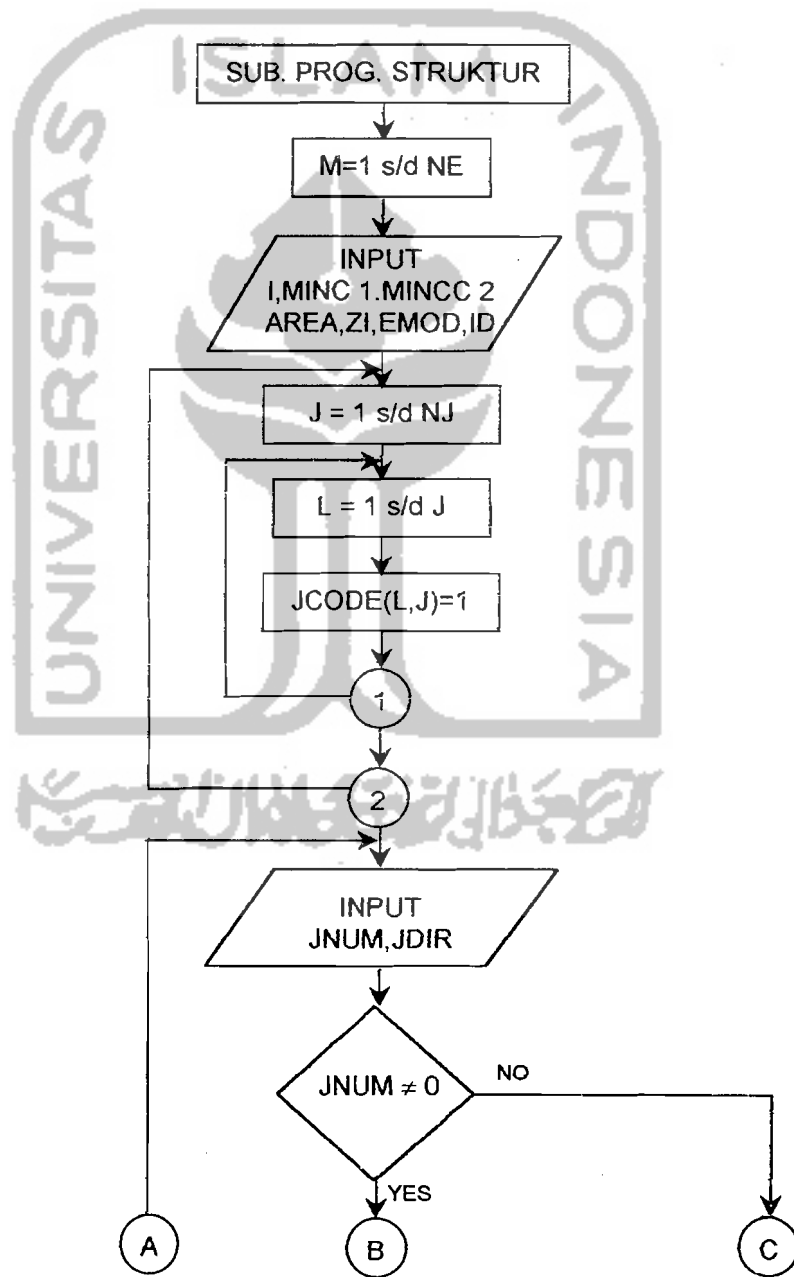
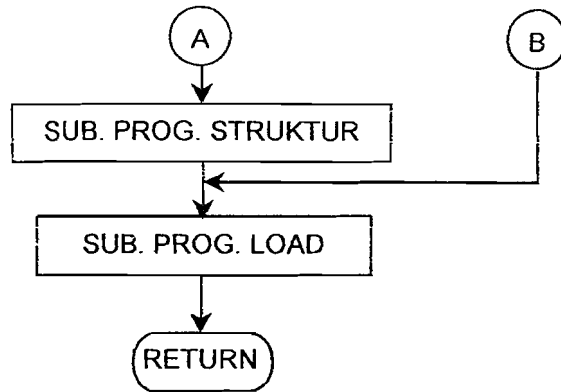
Terdiri dari gabungan beberapa Sub Program yang menjalankan program utama yang terdiri dari sub program Data, sub program System, sub program Result. Sub program utama terdiri dari beberapa sub program yang lain, sehingga dapat membentuk suatu struktur program seperti dapat dilihat pada gambar 3.2. di bawah ini.

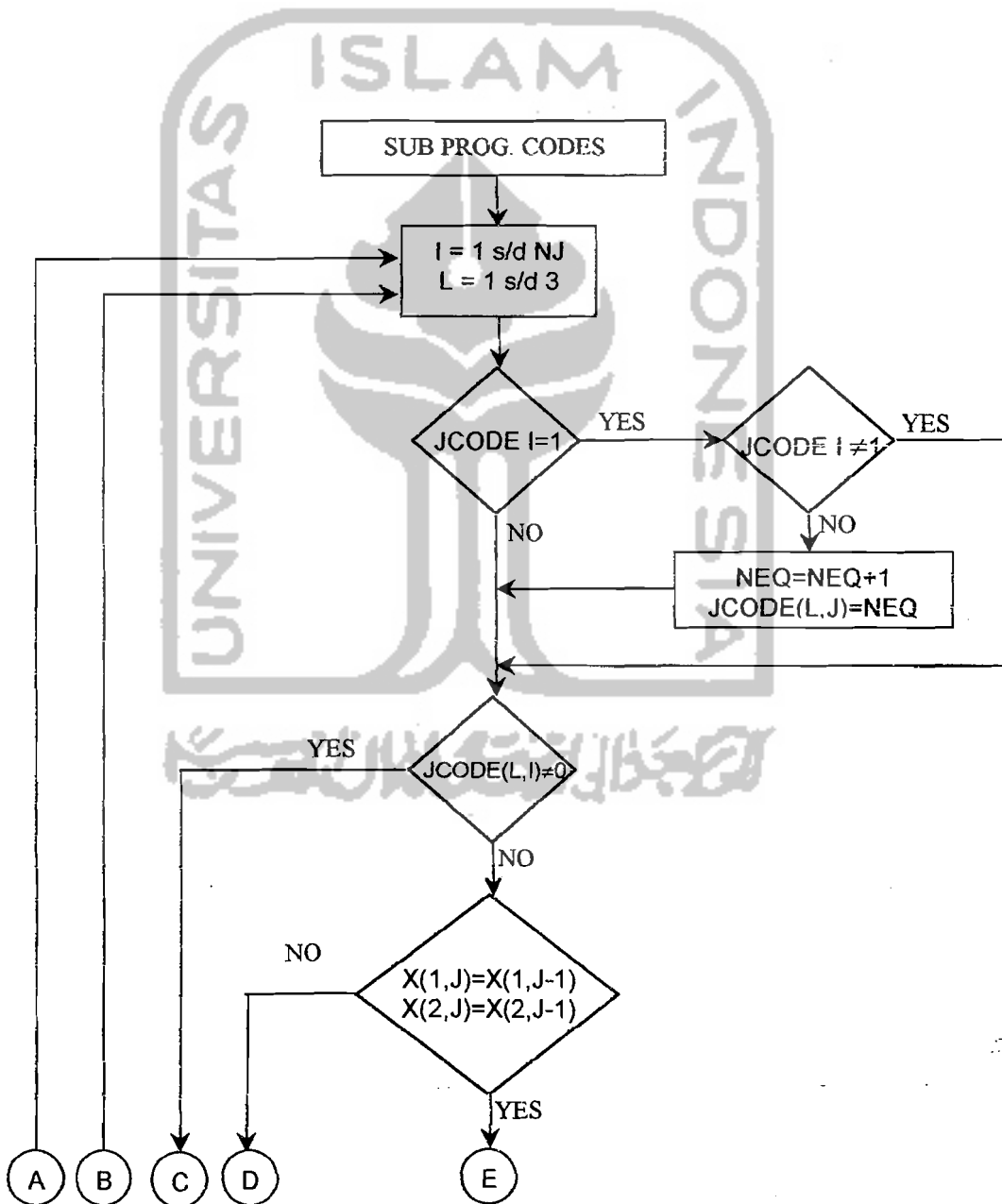
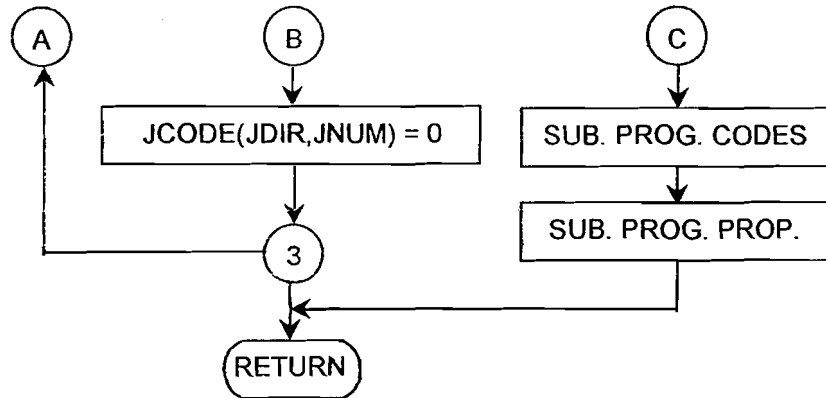


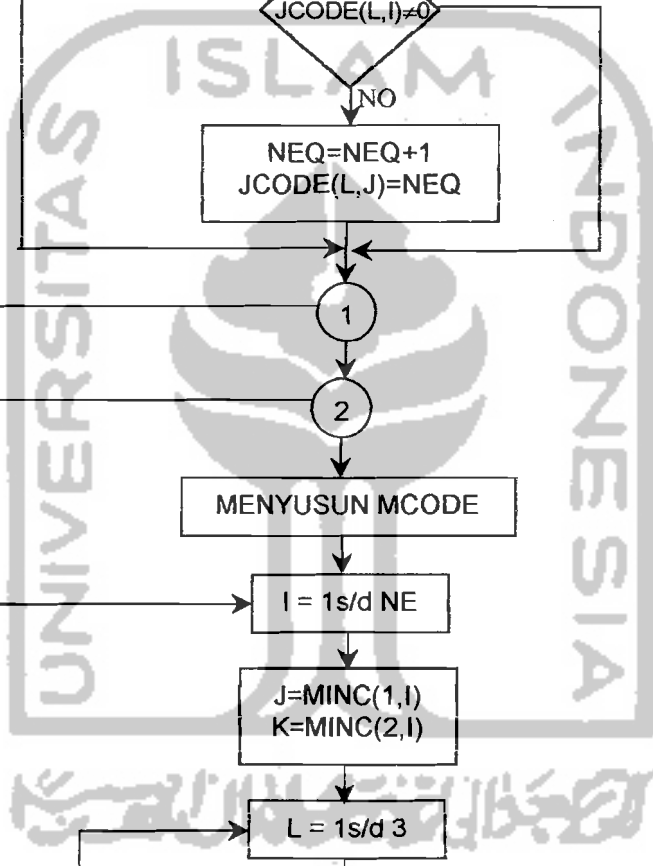
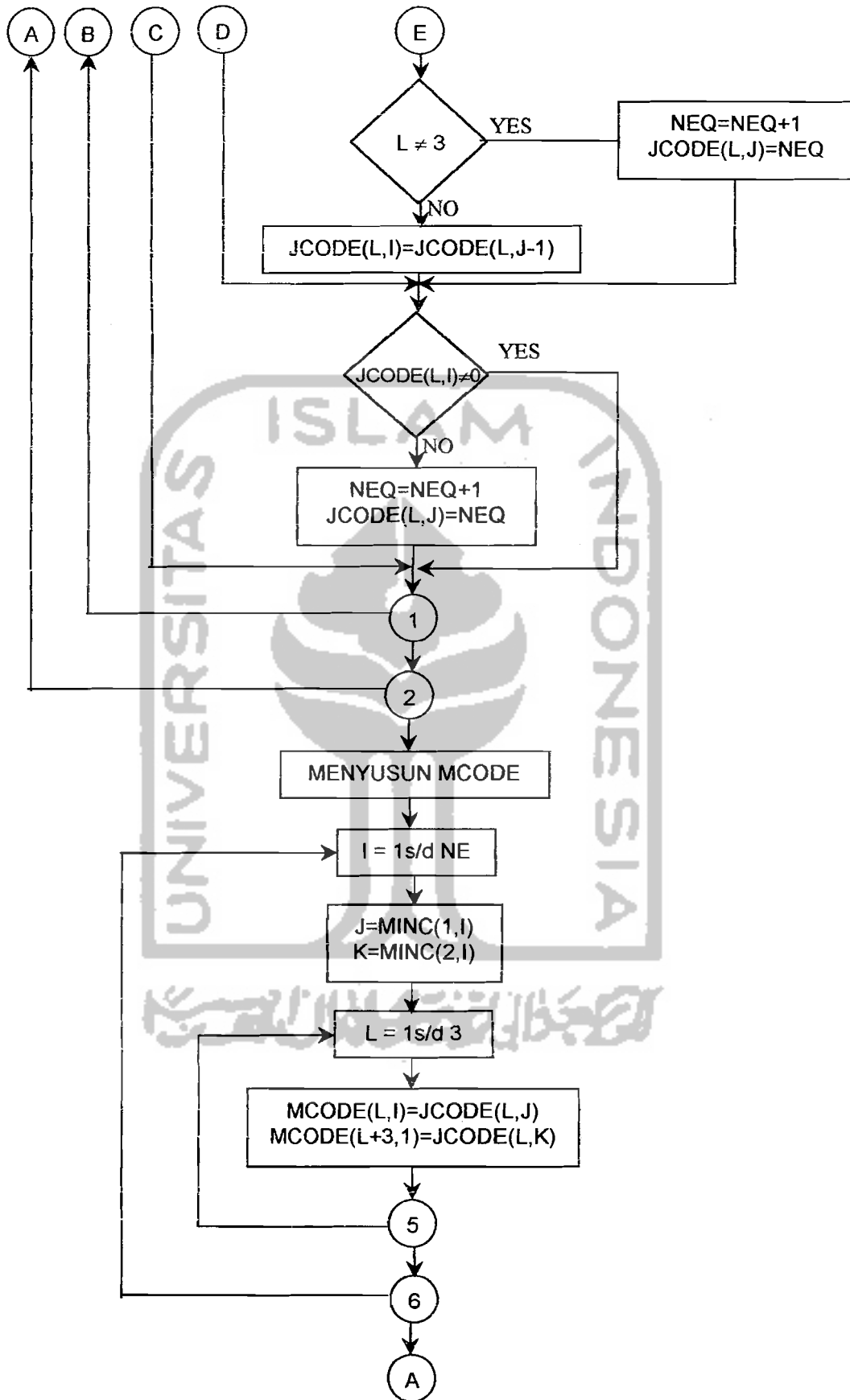
Gambar 3.2. Struktur Program

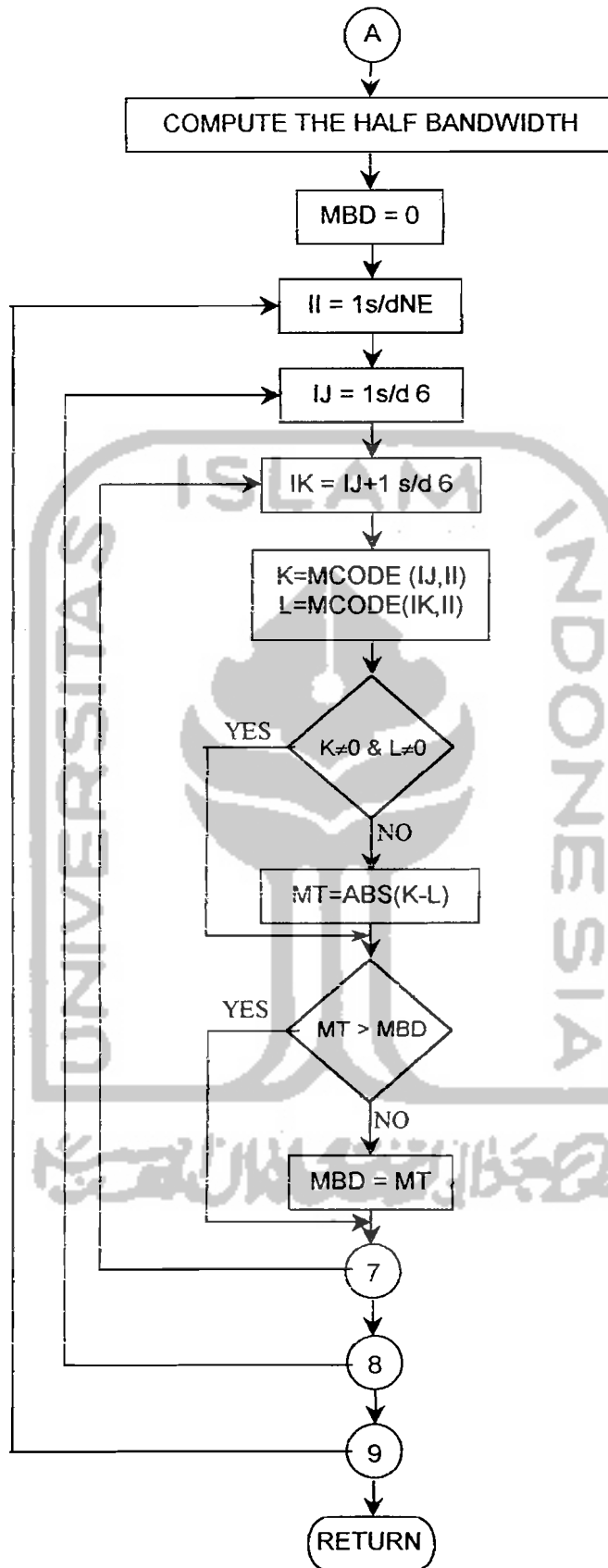
## 3.3.2. Flow Chart Program Utama

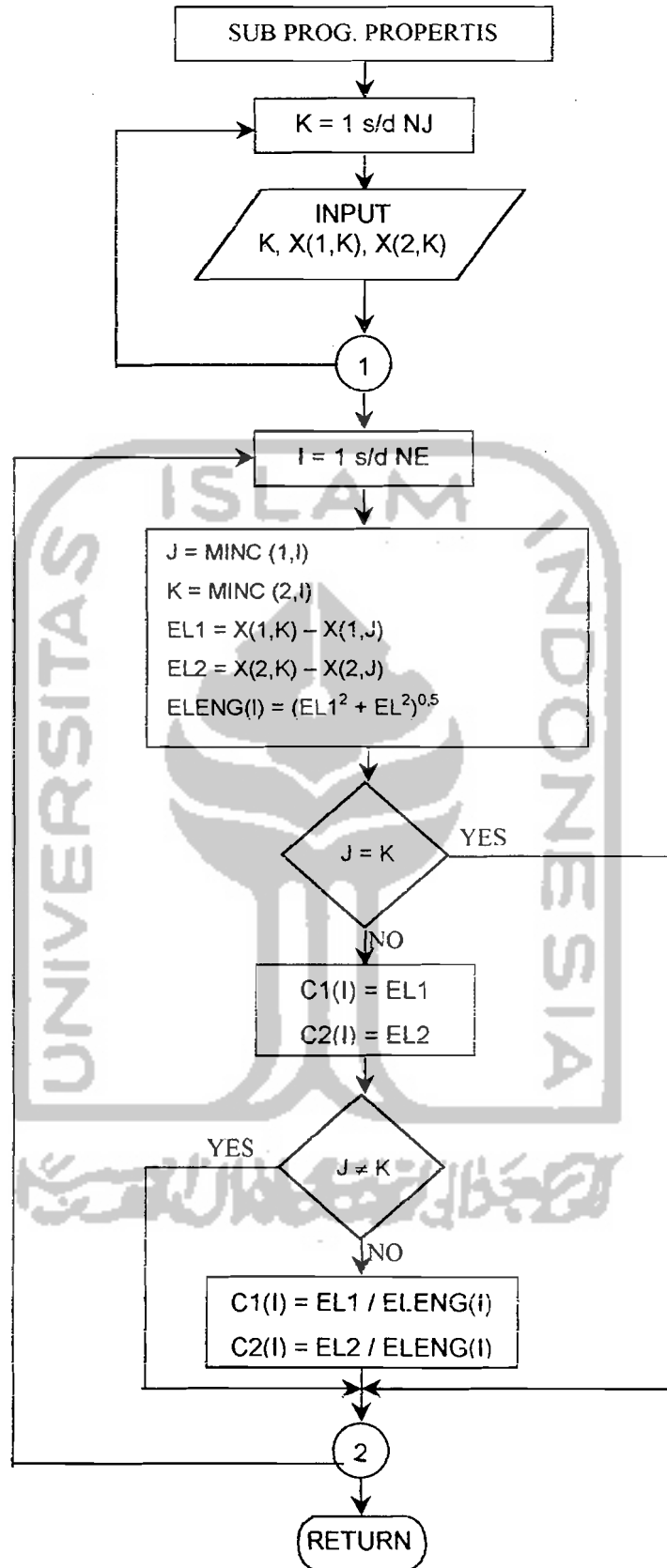


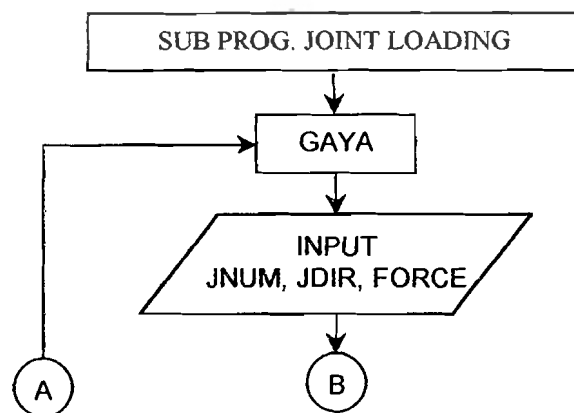
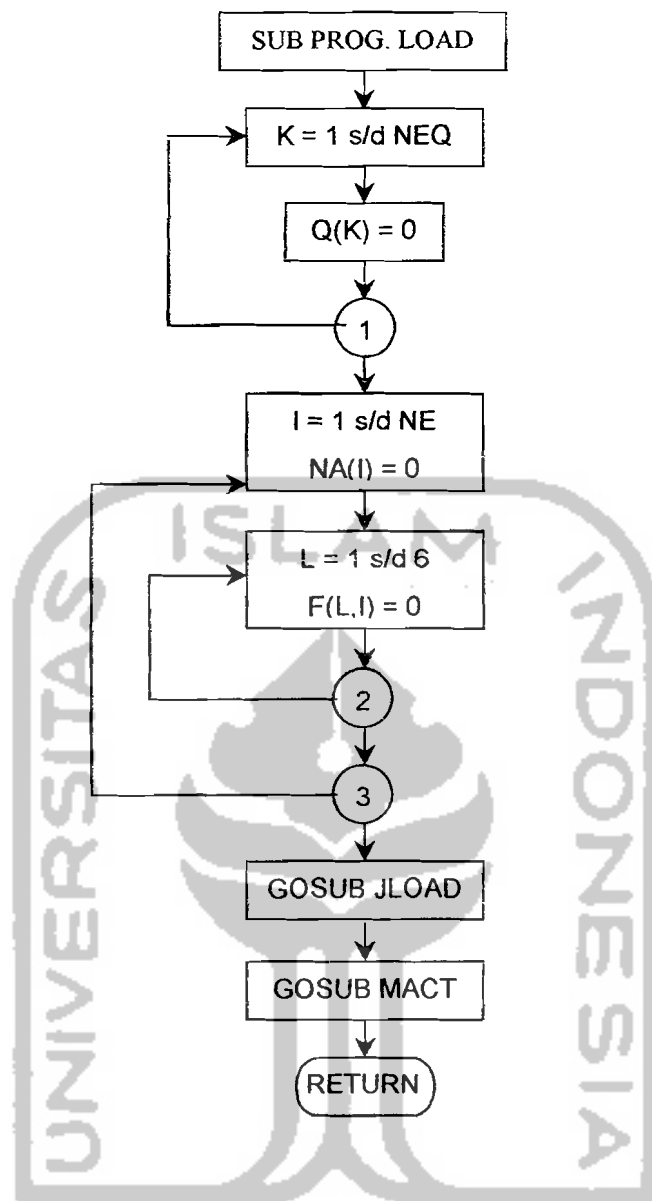




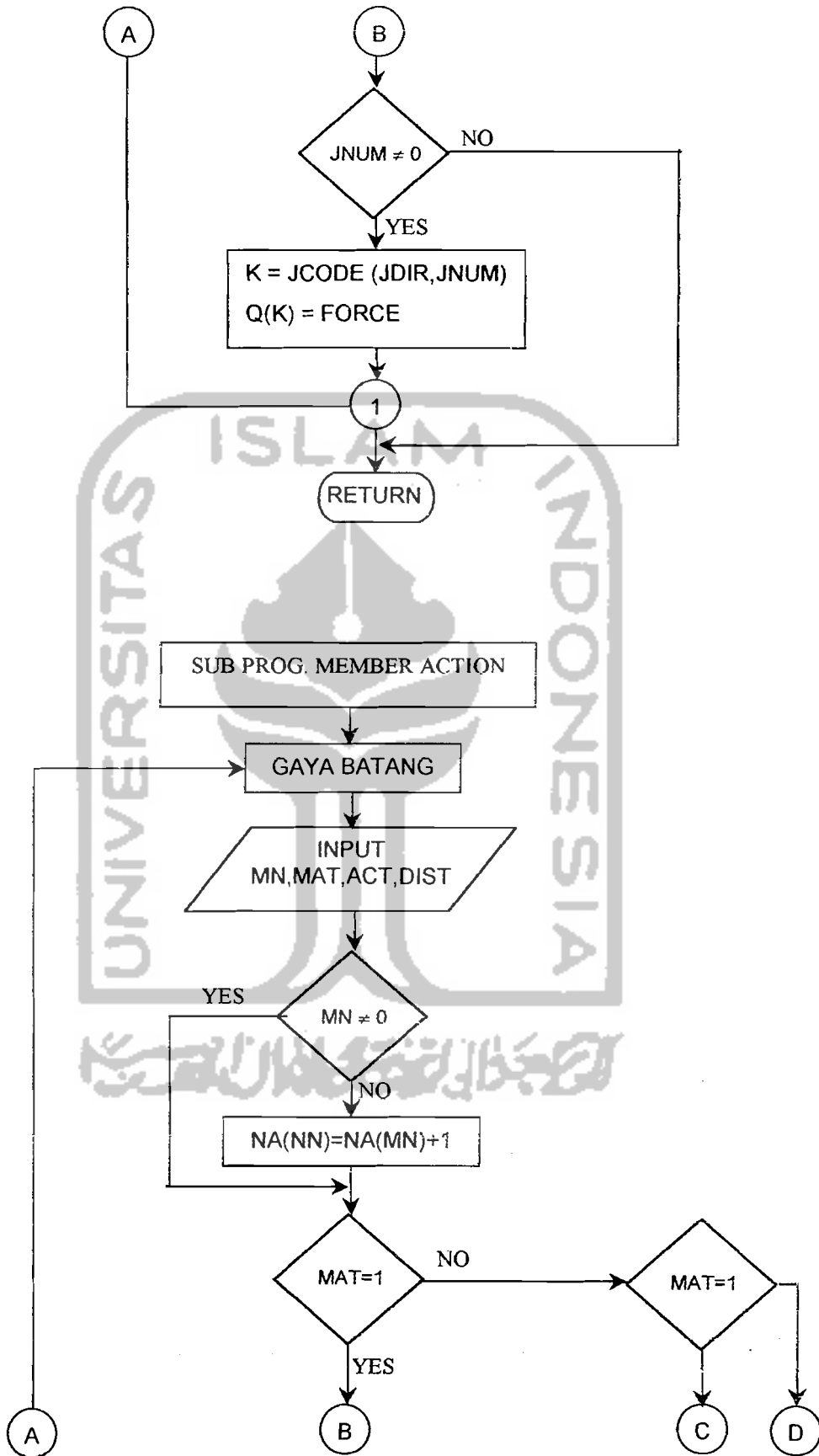


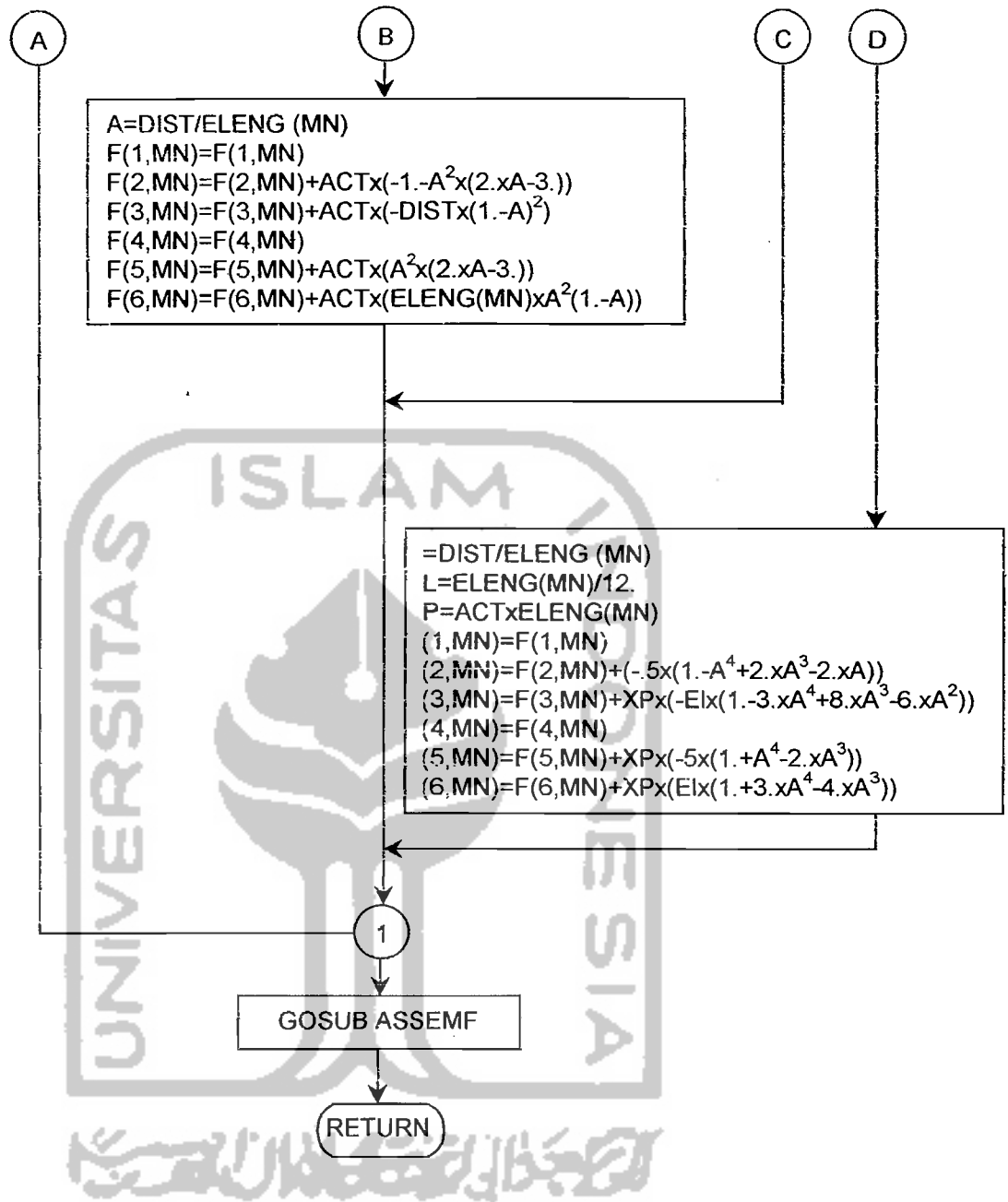








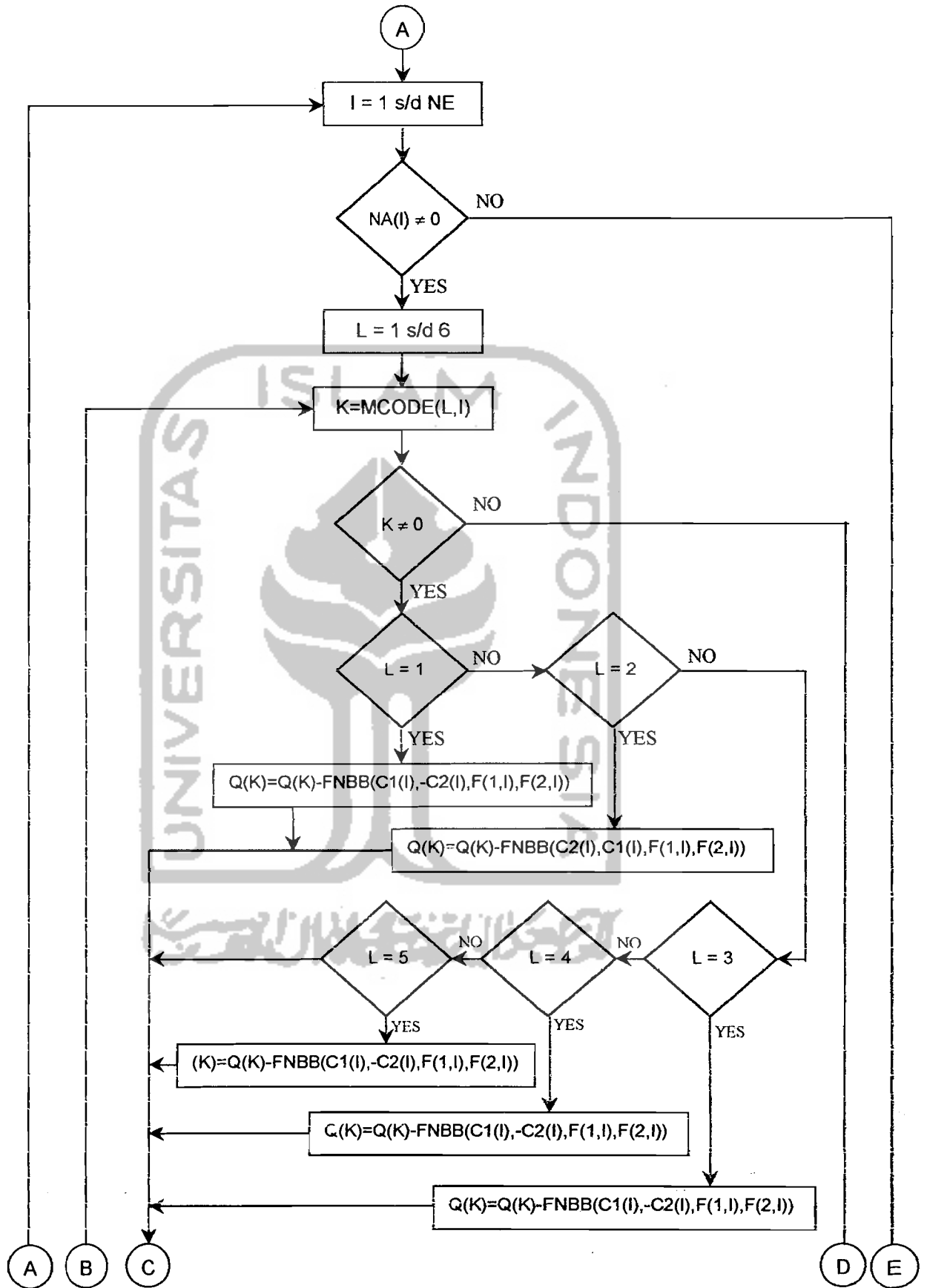


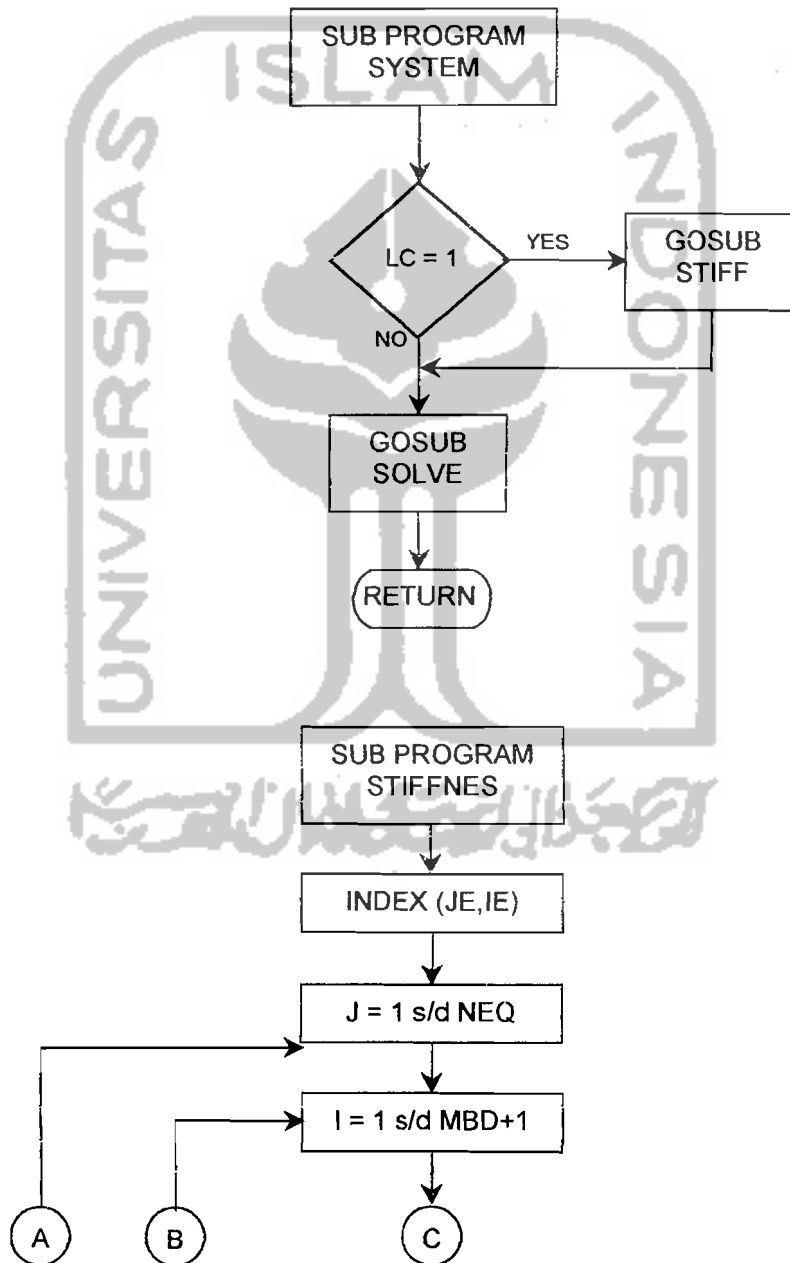
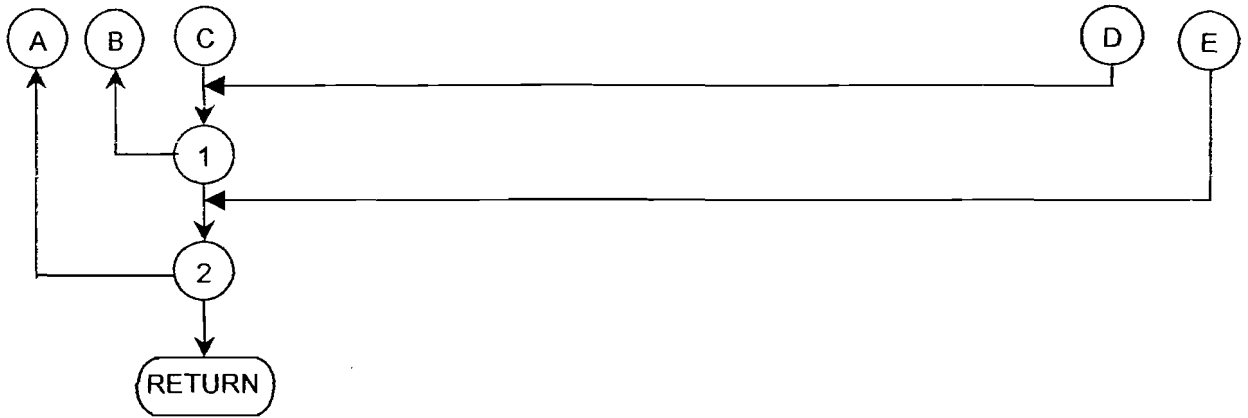


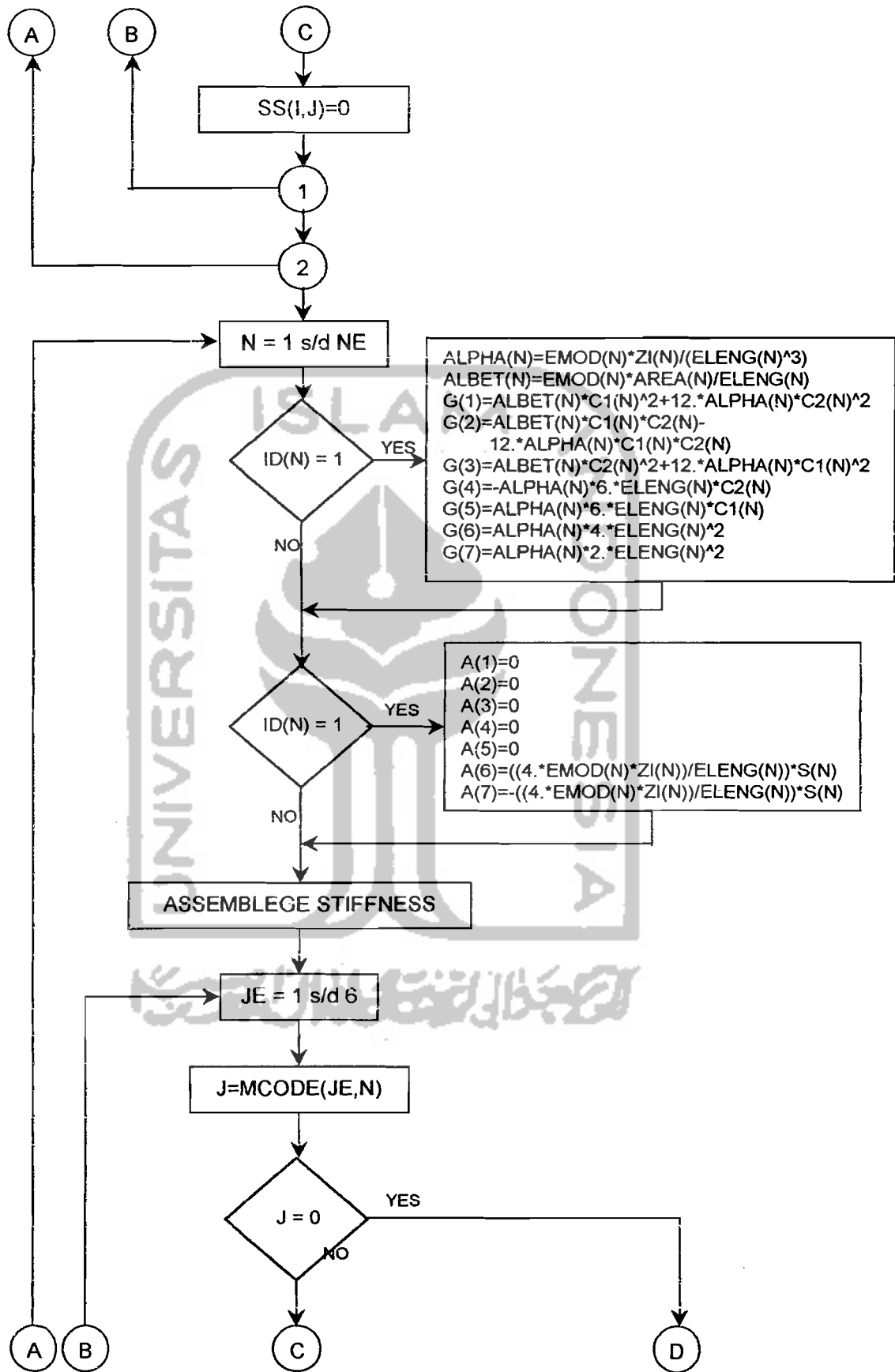
SUB PROG. ASSEMBLAGE FORCE MATRIKS

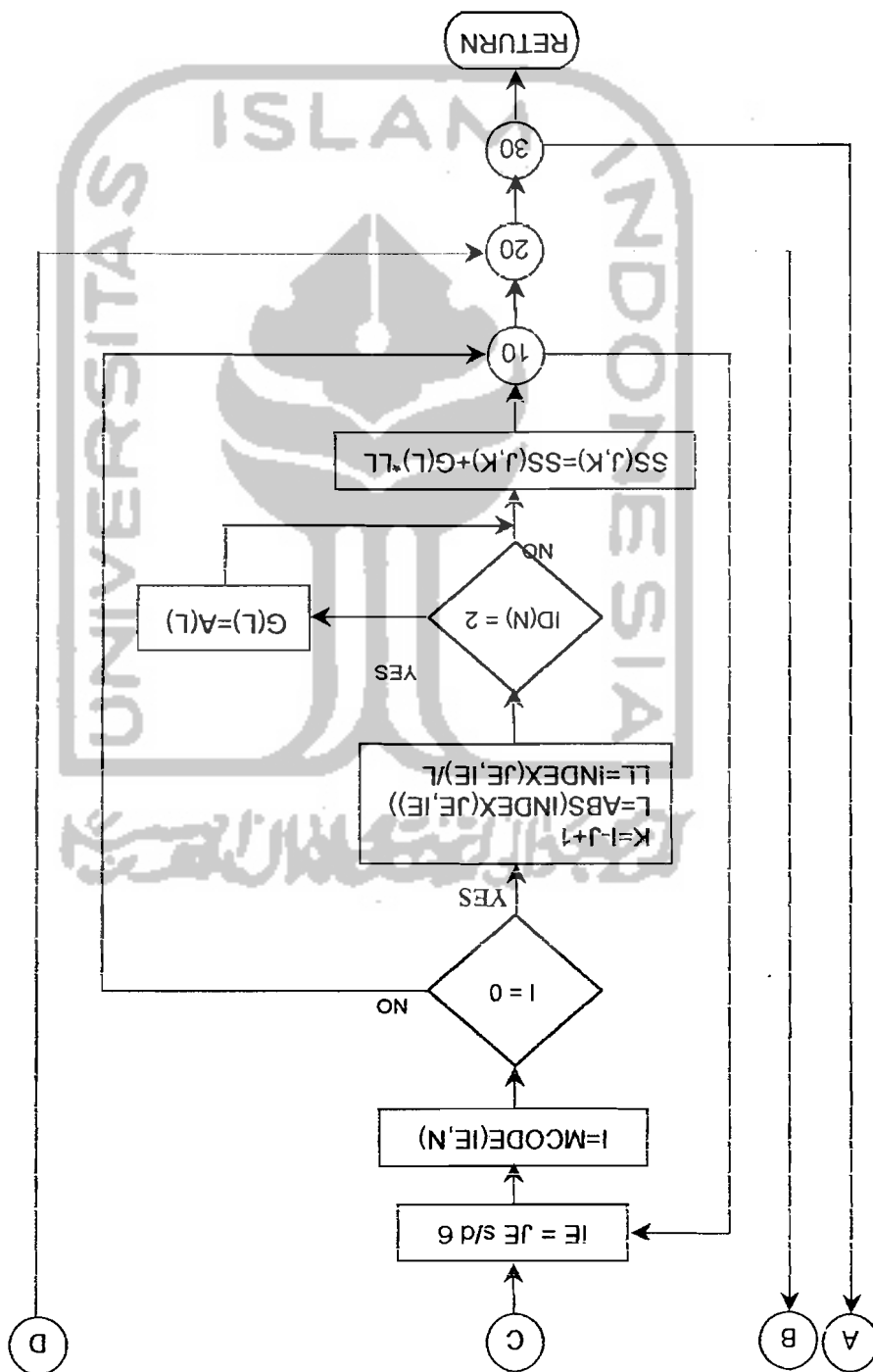
DEF FNBB(C1I,C2I,FLX,FLY  
FNBB=C1IxFLX+C2IxFLY

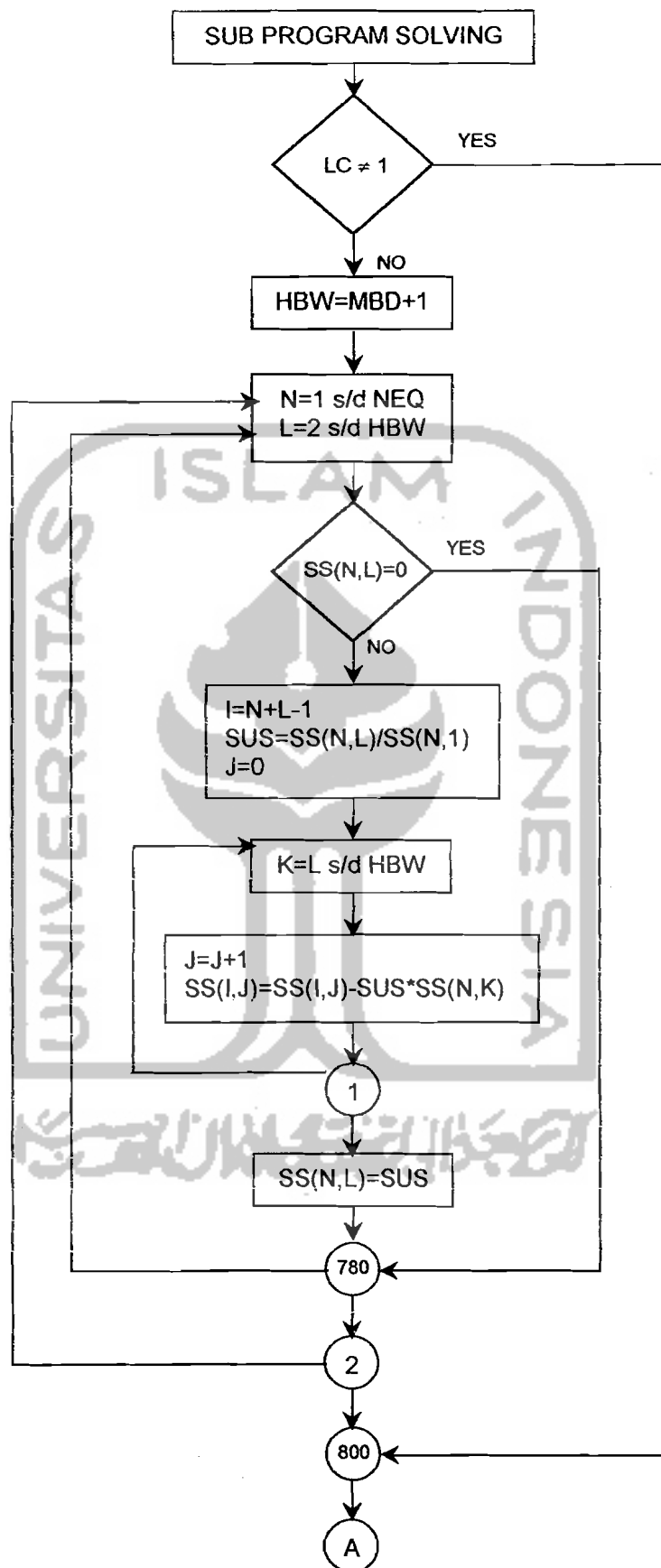


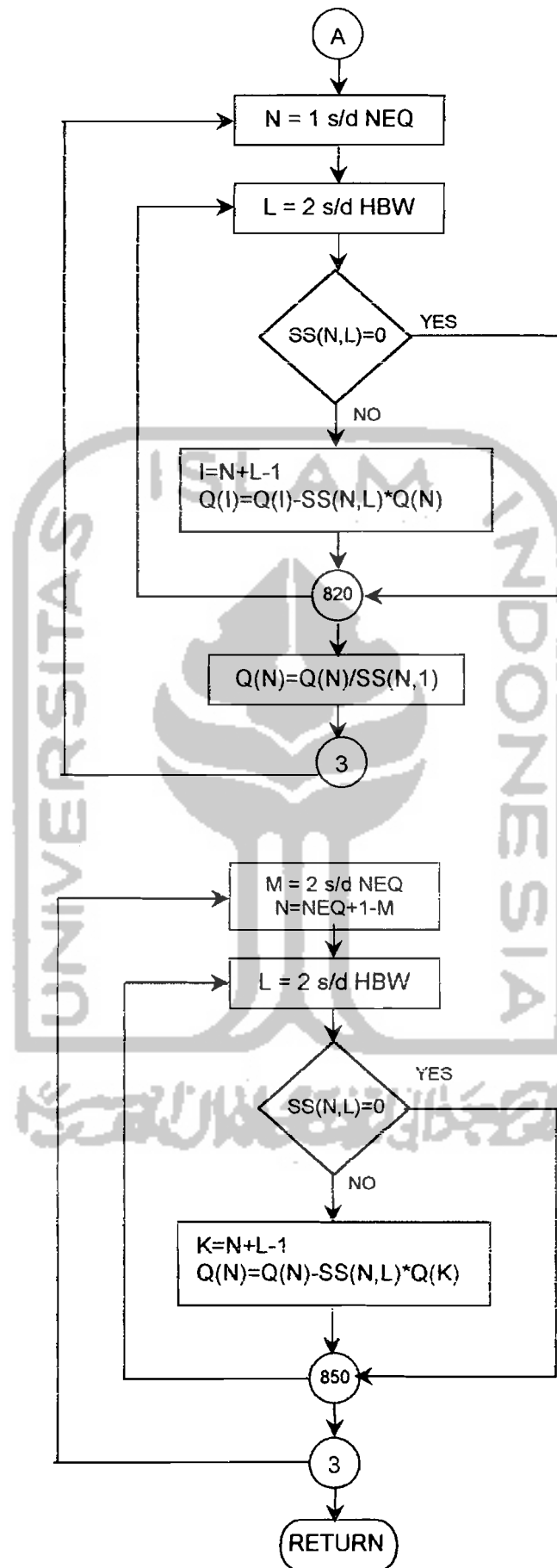




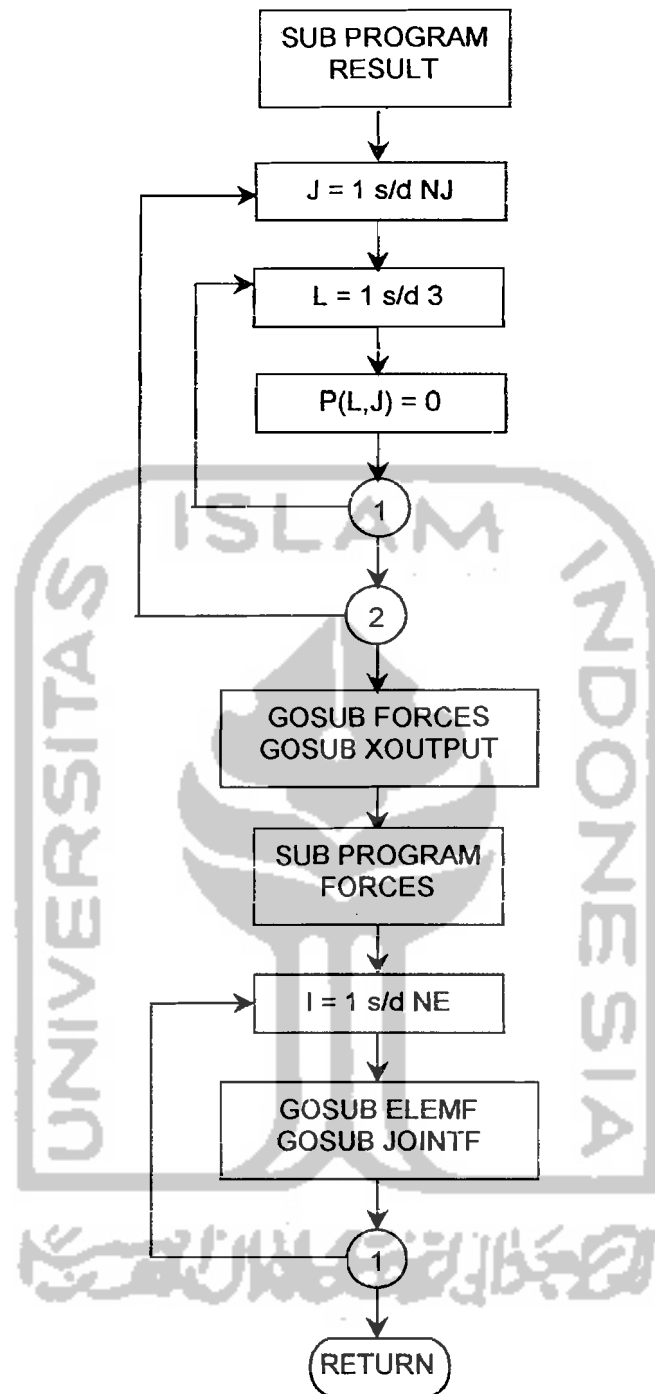


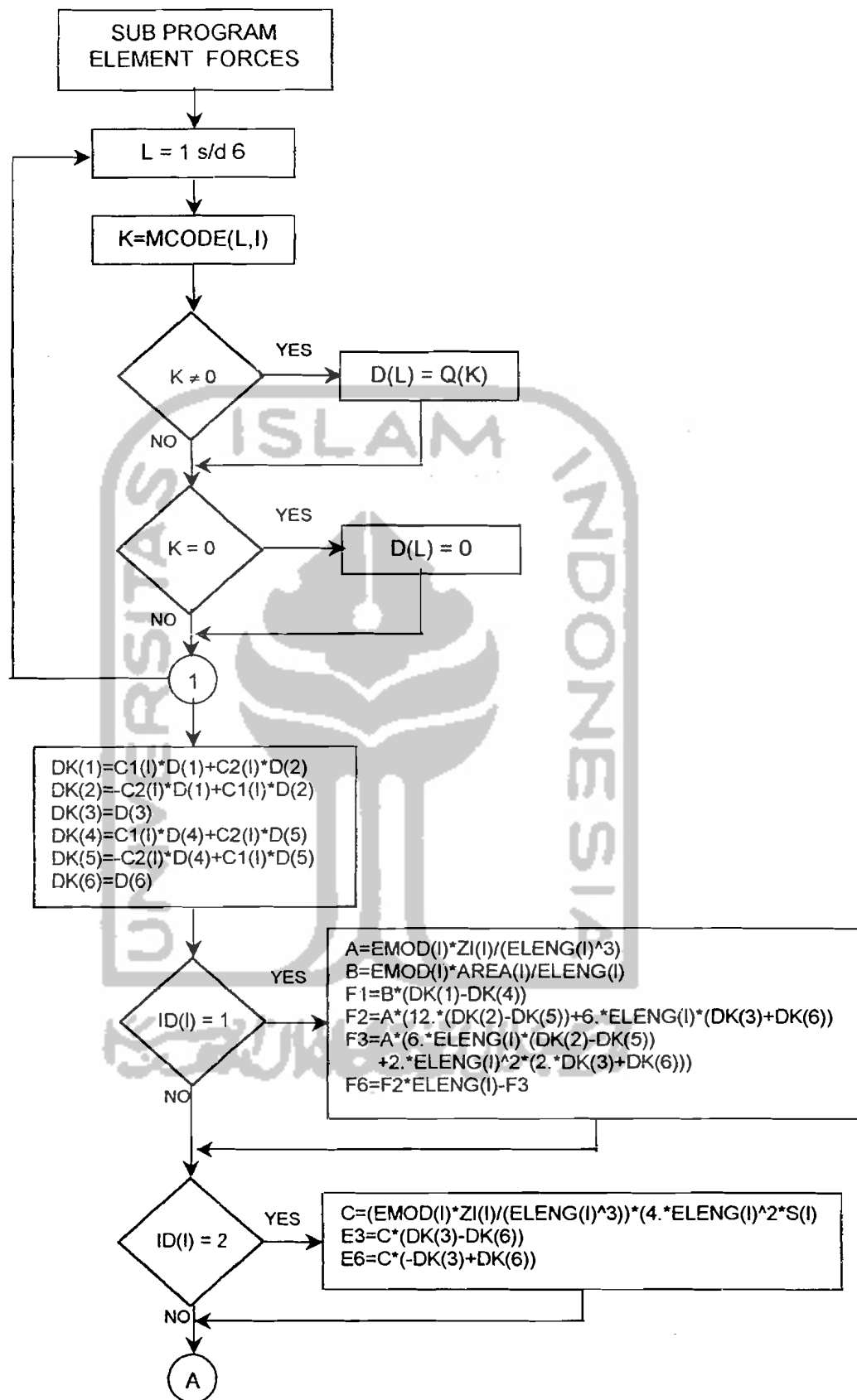


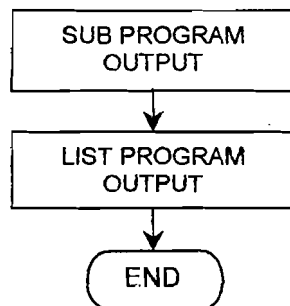
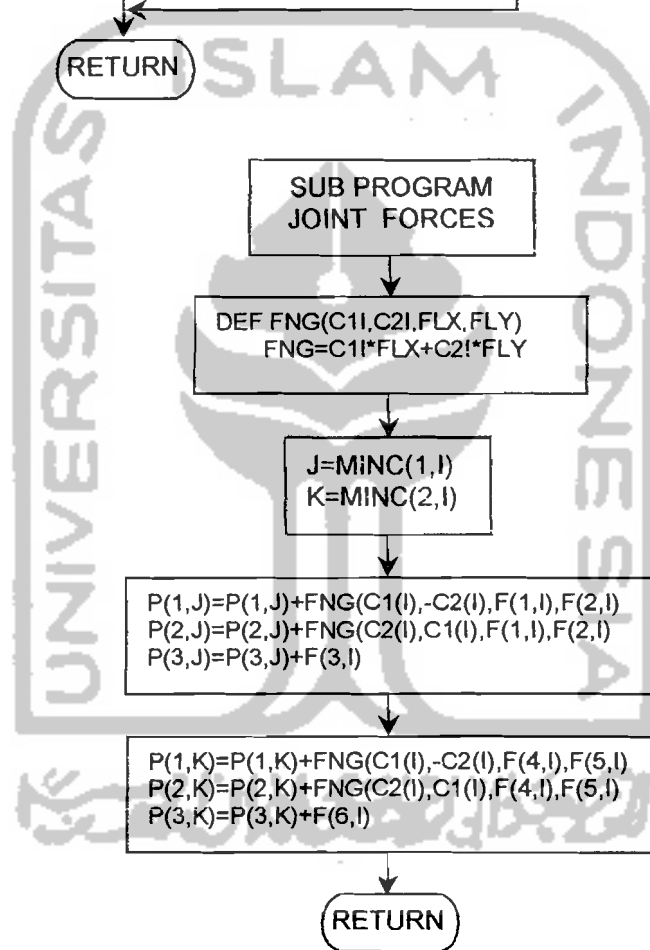
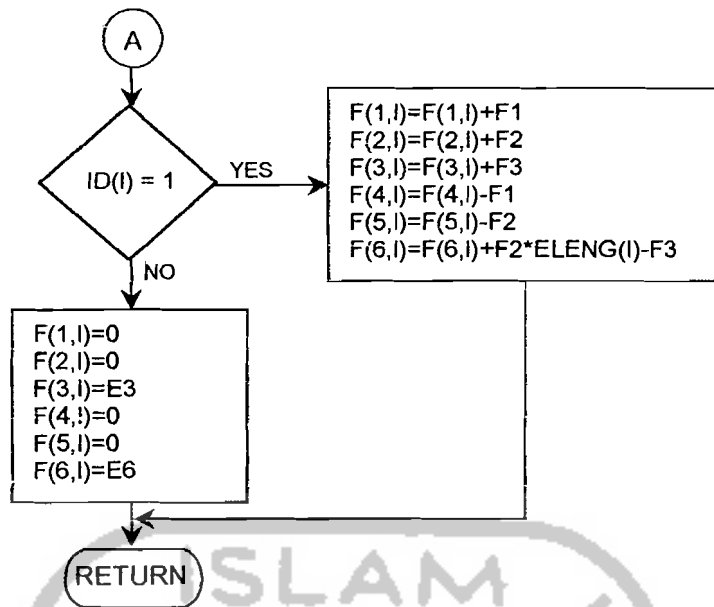


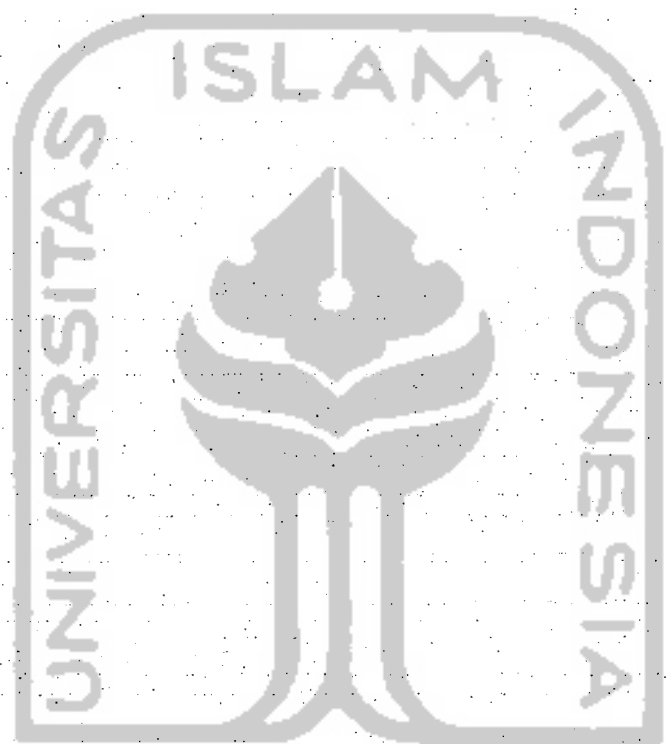












جامعة الإسلام في إندونيسيا