

PANDUAN PRAKTIKUM

DASAR PEMOGRAMAN KOMPUTER



PROGAM STUDI TEKNIK GEODESI
FAKULTAS TEKNIK
UNIVERSITAS PAKUAN
BOGOR

KATA PENGANTAR

Alhamdulillah, segala puji bagi Allah tuhan semesta alam, dengan izin dan kehendak-Nya Modul praktikum mata kuliah Dasar Pemrograman Komputer dapat diselesaikan. Sasaran dari Teknik Geodesi dengan dasar-dasar dalam praktikum ini, adalah untuk memperkenalkan mahasiswa pemrograman Komputer, mulai dari penusunan Algoritma, Flowchart sampai penulisan program.

Software yang digunakan dalam pelaksanaan praktikum adalah Bahasa Pemrograman Quick Basic. Alasan pemilihan Software ini adalah karena bahasanya sederhana, mudah dipelajari dan softwarenya mudah didapat sehingga akan membantu mahasiswa dalam mempelajarinya. Pada tiap materi praktikum disediakan lembar kerja untuk latihan mahasiswa yang ditandatangani oleh asisten, dengan demikian diharapkan Setiap mahasiswa benar-benar melakukan praktek untuk setiap yang telah disusun.

Tidak lupa penyusun ucapkan terimakasih yang sebesar-besarnya kepada saudara Hari Listkantyo ST. yang memberi pinjaman contoh materi tulisan ini, structural Jurusan Teknik Geodesi Fakultas Teknik Universitas Pakuan yang selalu mengingatkan pentingnya Modul ini, baik untuk kepentingan Mahasiswa, kepentingan penyusun dan kepentingan Jurusan. Juga kepada Sekretariat dan Laboran Jurusan Teknik Geodesi yang sangat banyak membantu.

Semoga modul ini bermanfaat, terutama bagi mahasiswa yang mengambil Matakuliah Dasar Pemrograman Komputer. Saran dan masukan untuk perbaikan modul ini dimasa datang sangat penyusun harapkan.






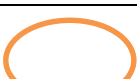
Bogor, Juli 2003

Penyusun

Ir. Marlis Yunanto

PRAKTIKUM I Alogaritma dan Flow Chart

Alogaritma program adalah urutan logika program dimulai dari starting awal, proses yang terjadi hingga hasil akhir dari program tersebut. Flow chart adalah diagram/skema yang menggambarkan aliran program yang diwakili oleh simbol-simbol untuk menyatakan aktifitas yang dilakukan. Untuk menggambarkan bagian alir tersebut digambarkan dengan simbol-simbol gambar tertentu. Simbol-simbol yang biasa digunakan adalah sbb:

| Simbol | Artinya | Keterangan |
|---|----------------------|--|
|  | Preparation | Menggambarkan awal suatu kegiatan |
|  | Terminator/ Terminal | Menggambarkan akhir suatu kegiatan atau berhentinya suatu program |
|  | Input/ Output | Menggambarkan input data yang dilakukan atau output data yang dihasilkan |
|  | Process | Menggambarkan proses data yang dilakukan |
|  | Decision | Menggambarkan adanya pilihan arah aliran program yang akan dituju |
|  | Row | Menggambarkan arah aliran program |
|  | Sambungan | Menunjukkan sambungan dari Flow Chart di halaman yang berbeda |
|  | Connector | Menunjukkan sambungan dari Flow Chart di halaman yang sama |

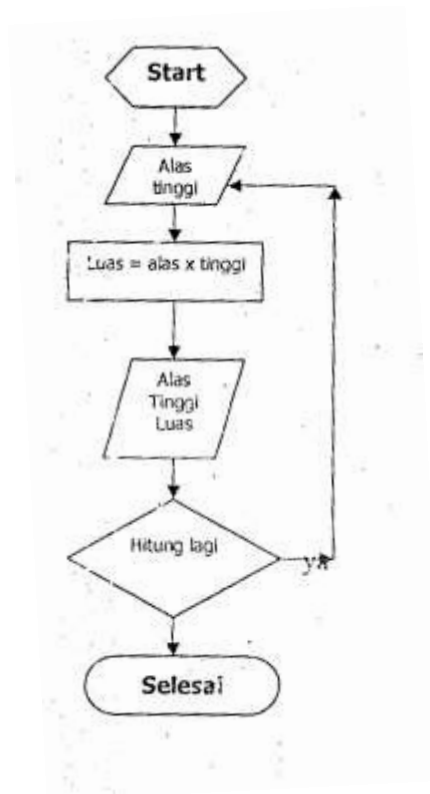
Contoh:

Akan dibuat program untuk menghitung luas segitiga.

a. Algoritma:

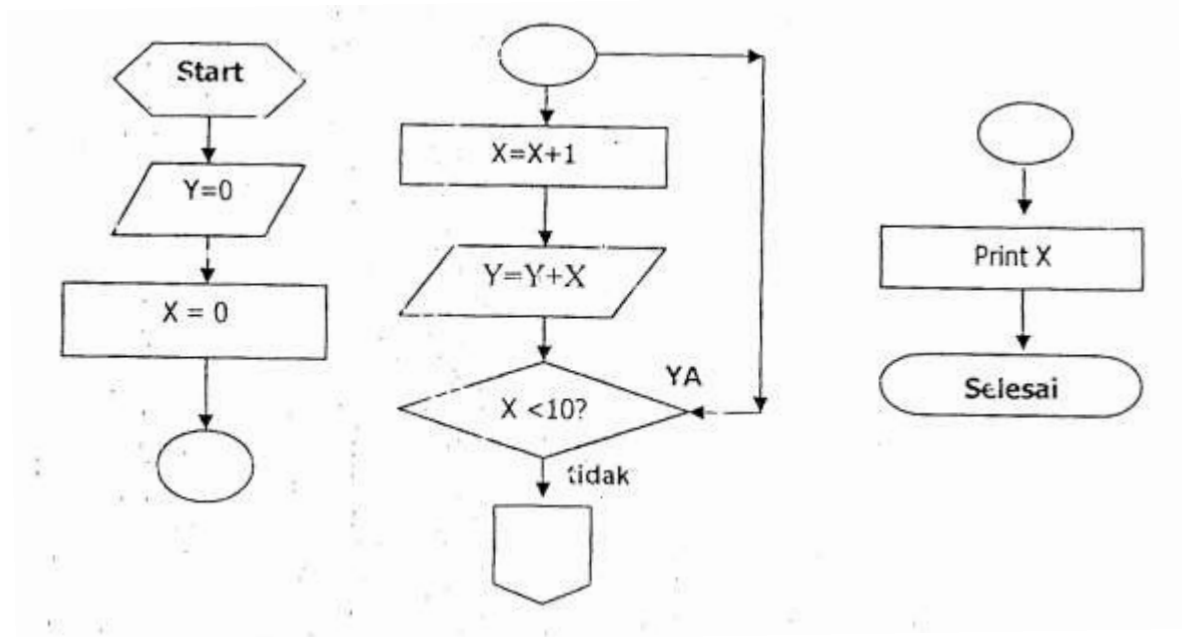
1. Program Mulai
2. Menulis input panjang alas dan tinggi segitiga
3. Menghitung luas segitiga $\text{luas} = (\text{alas} \times \text{tinggi})/2$
4. Menulis hasil hitungan luas
5. Meminta masukan apakah akan menghitung lagi
6. Jika iya kembali ke langkah 2
7. Jika tidak proses selesai
8. Program Selesai

b. FlowChart



Latihan untuk pemahaman (tulis dilembar kerja mahasiswa)

1. Carilah letak kesalahan pada simbol flowchart berikut



2. Buatlah algoritma flowchart untuk menghitung akar persamaan kuadrat

RU : $X_1 X_2 = (-b \pm \sqrt{D}) / (2a)$

$D = b^2 - 4 \cdot a \cdot c$

Jika $D > 0$ maka $X_1 \neq X_2$

Jika $D = 0$ maka $X_1 = X_2$

Jika $D < 0$ maka X_1 dan X_2 adalah bilangan imajiner

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM II Identifier (Konstanta dan Variabel)

Identifier Adalah karakter atau rangkaian karakter yang mewakili suatu obyek. Hal ini diperlukan untuk mengenali obyek-obyek yang akan kita gunakan. Identifier yang memiliki nilai konstan disebut konstanta sedang yang memiliki nilai yang dapat berubah disebut dengan variabel. Penulisan identifier harus sesuai dengan aturan bahasa pemrograman yang digunakan. Dalam Bahasa Pemrograman Quick Basic aturan Penulisan identifier adalah sebagai berikut :

- Karakter yang diperbolehkan adalah karakter standar (a-z, A-Z, 0-9)
- Minimal 1 karakter
- Karakter awal harus merupakan karakter alpha numeric (huruf)
- Bukan kata yang ada dalam library Qbasic (read, end, goto, dll)
- Tidak ada karakter blank “ “

Contoh :

- Luas
- X1
- A
- Rata-rata
- Sudut
- Dll

Selain itu dalam mendefinisikan nilai dari suatu variable/constant harus sesuai dengan type data dari masing~masing Variable/Constanta. Untuk mendefinisikan type data dari variabel menggunakan perintah (command) DIM.

Contoh:

DIM luas AS single -> type data bilangan real

DIM nama AS string -> type data textual

DIM data AS integer -> type data bilangan integer

Latihan

Penulisan constanta dan variable sesuai dengan type datanya.

```
1. cls
rem memasukkan nilai panjang dan lebar
input " Nilai Panjang = "; Panjang
input " Nilai Lebar = "; Lebar
rem menghitung nilai Luas dan Keliling
Luas = panjang * lebar
Keliling = 2*(panjang+lebar)
rem memuliskan hasil hitungan dilayar
print " Nilai Luas = "; luas
print " Nilai Keliling = "; keliling
end
```

Perhatikan program diatas, ada berapa variabel yang digunakan

```
2. cls
rem penggunaan Variabel String
DIM STR1 AS STRING
rem penggunaan Variabel String dengan pembatasan
DIM Str 2 AS STRING * 6
STR1 = "1234567890"
STR2 = "1234567890"
Print STR1, Str2
```

Jalan program diatas, beri uraian mengapa hasil str1 tidak sama dengan str2

```
3. cls
rem penggunaan Variabel String, Single, Integer
cls
DIM Num AS integer
DIM Time AS single
DIM Grade AS string
DIM sum AS single
Print
Grade = A
Time = 20.15
Num = 10.35
Time + num = Sum
Print
Print "sum = "; Sum
```

Jalan program diatas, perbaiki jika atau logika program salah, beri uraian mengapa demikian.

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM III Operator Matematika

Penulisan lambing Operator matematika dalam bahasa pemograman Quick Basic sedikit berbeda dengan operator matematika biasa terutama untuk perkalian, perpangkatan, akar kuadrat. Demikian juga dengan nilai sebuah variable, dalam Quick Basic variable yang sama dalam satu kalimat matematika (satu progam) dapat memiliki nilai berbeda.

Contoh :

$$B = B + A$$

Misal jika nilai input adalah $A = 1$, $B = 2$, maka akan menghasilkan nilai output $B = 3$.

Simbol yang digunakan dalam Quick Basic :

- Operator Arimatika

| Lambang | Arti | Contoh |
|---------|---------------------|-------------------------|
| + | Penjumlahan | $C = A + B$ |
| - | Pengurangan | $E = D - C$ |
| * | Perkalian | $Z = X * Y$ |
| / | Pembagian | $T = X/Y$ |
| \ | Pembagian (integer) | $K = M \setminus N$ |
| ^ | Pangkat | $Y = X^3$ |
| Sqr (X) | Akar | $A = \text{Sqr} (X)$ |
| MOD | Modulus | $D = X \text{ MOD } 10$ |

- Contoh penggunaan simbol dalam pemograman

| Ekpresi Aljabar | Ekspresi Quick Basic |
|---------------------------|----------------------|
| $\frac{a + b^2}{a - b^2}$ | $(a+b^2)/(a-b^2)$ |
| $\frac{x + y}{u + w/a}$ | $(x+y)/(u+w/a)$ |

Latihan untuk pemahaman :

1. Perhatikan penulisan operasi matematika dalam bahasa Quick Basic dibawah ini

$$X = a^b + c * d/b + b*a - c$$

$$Y = a ^ (b + c) * (d / b) + b * a - c$$

$$Z = a ^ (b + c) * d / b + b * (a - c)$$

$$Q = (a ^ b + c) * d / (b + b) * a - c$$

- Tulis dalam bahasa matematika biasa
- Hitung X, Y, Z dan Q secara manual jika $a = 5$, $b = 2$, $c = 3$, $d = 7$
- Eksekusi progam diatas, samakan hasil hitungan manual anada dengan hasil hitungan computer
- Tuliskan urutan proses berdasarkan hirarkinya

2. Tulis pernyataan matematika berikut dalam bahasa Quick Basic :

a. $X_1 = \frac{-b\sqrt{b^2-4ac}}{2a}$

b. $V \text{ tabung} = \pi.r^2.T$

c. $X_2 = X_1 + DSin\alpha$

d. $X_1 = \frac{x^{n+1}}{y^{n+1}}$

e. $Z = \frac{2x^3-y^2}{x^2-2y^2}$

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM IV Pemograman Sederhana

Statement yang ada dalam quick basic

| | |
|-------|---|
| Cls | Untuk membersihkan layar |
| Rem | Membuat apa yang tertulis sesudahnya diabaikan compiler |
| Input | Untuk membaca data dari "keyboard" |
| Read | Membaca data dari berkas atau statement data |
| Data | Untuk menyimpan nilai variabel dalam program |
| Print | Untuk mencetak/mendisplay informasi di monitor |
| End | Akhir program |
| DLL | |

Contoh program sederhana,

```
1. Cls
  Rem statement read, data
  Read x, y, z
  A = (x*y)/z
  B = (x^y)*z
  C = A+B
  Print "Nilai A = "; A
  Print "Nilai B = "; B
  Print "Nilai C = "; C
  End
  Data 5, 2, 7
2. cls
  rem I/O untuk data string
  INPUT "NAMA MAHASISWA :";NAMA$
  INPUT "N R P      :";NRP$
  READ STR1$, STR2$
  PRINT
  PRINT "Nama :";UCASE$ (NAMA$)
  PRINT "NRP  :";UCASE$ (NRP$)
  PRINT LCASE$ (str1$) + LCASE$ (STR2$)
  DATA "Program Studi","TEKNIK sipil"
  End
```

Perhatikan hasil yang diberikan setiap statement

Latihan untuk pemahaman

Buat program untuk menghitung panjang A – B, dilengkapi dengan inputi data pembuat program.



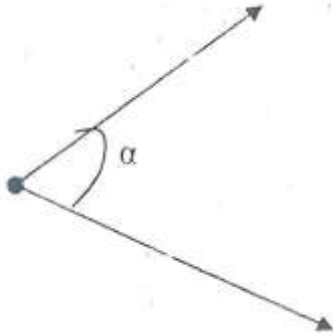
Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM V Komputasi Trigonometris



Sudut α dihitung dengan mencari selisih antara bacaan arah B dengan arah A dan B sendiri akan terbaca dalam bentuk dd°mm'ss", sehingga harus diubah dulu menjadi derajat. Untuk menghitung sudut α , didalam bahasa pemograman quick basic nilai α harus diubah menjadi radian. Cara mengubah nilai α dalam format dd°mm'ss" menjadi radian adalah sebagai berikut :

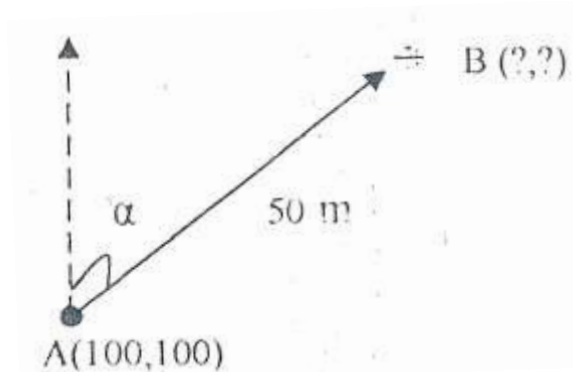
1. dd°mm'ss" diubah terlebih dahulu menjadi $dd + mm'/60 + ss''/3600$
2. Kemudian hasilnya dikalikan phi, phi = 3.14

Contoh program

```
cls
sudut = 45 + 32/60 + 30/3600
phi = 3.14
R = phi/180
A = sudut*R
B = sin (a)
Print "A = " ; A
Print "Sin (A)= " ; B
End
```

Latihan pemahaman

Suatu titik A diketahui koordinatnya (100,100). Suatu titik B yang berjarak 50 m dari titik A akan dicari koordinatnya. Ulis progamnya (azimuth A-B = 45°30'20")



Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM VI Statement Kendali GOTO

Tatemen kendali digunakan untuk mengubah urutan eksekusi program yang berurutan menuju ke proses yang berada ditempat lain baik proses yang berada pada baris sebelumnya atau melangkah kebaris sesudahnya. Meskipun efeknya dari statemen ini membuat eksekusi program menjadi tidak berurutan tetapi pemakaian statemen ini tidak bisa dihindarkan.

Contoh :

```
1. Cls
  Rem statemen kendali goto
  10'awal program
  Input "Masukan Nilai x = "; x
  Input "Masukan Nilai y = "; y
  Print "Nilai x adalah = "; x
  Print "Nilai y adalah = "; y
  Input "Hitung Lagi? (Y/T) "; y$
  If y$ = "y" goto 10
  Print
  Print "Terima Kasih"
  end
2. rem menghitung nilai rata-2
  dim x as single
  dim xx as single
  sim rtx as single
  sim n as integer
  rem mendefinisikan nilai awal
  xx = 0
  rtx = 0
  n =
10 rem memasukan data
  input "masukan nilai X = ";x
  n = n+1
  xx = xx+x
  rts = xx/n
  input "masih ada data? (y/n)"; y$
  if y$ = "y"
  goto 10
  rem output data
  print "Jumlah Nilai = "; xx
  print "Jumlah Data = "; n
  print "Nilai rata-rata = "; rtx
  end
```

Jalankan kedua program diatas, beri penjelasan bagaimana proses terjadi.

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM VII Statement Kendali IF..THEN..ELSE

Statement kendali IF THEN ELSE adalah statemen untuk memberi pilihan arah eksekusi program sesuai dengan pilihan (logika) yang diberikan. Berikut adalah sebuah program untuk menghitung nilai persamaan kuadrat. (Flowchart pada praktikum 1 dan 2).

| | |
|--|--|
| $X_1 X_2 = \frac{-b \pm \sqrt{D}}{2a}$ $D = b^2 - 4. a. c$ | Jika $D > 0$, maka $X_1 \neq X_2$ Jika $D = 0$, maka $X_1 = X_2$ Jika $D < 0$, maka X_1 dan X_2 bilangan imajiner |
|--|--|

Contoh program akar-akar persamaan kuadrat

```

cls
Rem mendefinisikan variable
DIM x AS SINGLE
DIM x1 AS SINGLE
DIM x2 AS SINGLE
DIM xi AS STRING
REM membaca data a, b, c
INPUT "A="; a
INPUT "B="; b
INPUT "C="; c
REM proses menghitung D
d = b^2 - 4*a*c
REM statement kalang, dengan logika
IF d > 0 THEN
    x1 = (-b + SQR (d))/(2*a)
    x2 = (-b - SQR (d))/(2*a)
    PRINT USING "X1 =###.##"; x1
    PRINT USING "X2 =###.##"; x2
ELSE
    REM jika d = 0 maka x1=x2=x
    IF d = 0 THEN
        x = -b/(2*a)
        PRINT USING "X1=X2=X=###.##"; x
    ELSE
        REM jika d<0 maka x adalah imajiner
        xi = "bilangan imajiner"
        PRINT : PRINT "X adalah"; xi
    END IF
END IF
END
    
```

1. Coba dengan menggunakan $a = 8, b = 8, c = 4$
2. Coba dengan menggunakan $a = 8, b = 8, c = 2$
3. Coba dengan menggunakan $a = 4, b = 8, c = 2$

Beri uraian tentang hasil eksekusi program diatas

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUMVIII Statement Kendali FOR...NEXT

Statemen kendali FOR...NEXT adalah statemen untuk melakukan pengulangan proses.

Berikut adalah contoh program yang menggunakan statemen kendali FOR...NEXT.

Tuliskan program ini, eksekusi dan perhatikan efek pada setiap langkah

```
1. CLS
   A = 2
   B = 0
   For I = 1 TO 10 STEP 2
   A = A + 1
   B = B ^ 2
   PRINT "I ="; I, "A ="; A, "B ="; B
   NEXT I
2. REM statement FOR...NEXT dua lapis
   CLS
   A = 2
   B = 0
   PRINT " I J A B"
   PRINT " _ _ _ _ _ "
   FOR I = 1 TO 3
       FOR J = 1 TO 3
           A = A + 1
           B = A ^ 2
           PRINT I; J, A, B
       NEXT J
   NEXT I
   PRINT " _ _ _ _ _ "
   END
3. Rem FOR...NEXT untuk statement read...data
   For I = 1 to 5
   Read x
   y = x^2+x+2
   print I,x,y
   next i
   end
   data 2,5,7,8,1,9,5
```

Latihan untuk pemahaman

Buatlah program untuk transfer data suhu dari 0 – 100°C dengan setiap step 10° ke derajat Reamur dan Fahrenheit.

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM IX SUBROUTINE

Subroutine/subprogram dibuat untuk menyingkat penulisan program. Penulisan subroutine terpisah dari program utama. Penulisan statemen GOSUB harus selalu diikuti paling sedikit 1 statemen RETURN. Demikian juga RETURN harus diikuti statemen GOSUB.

Contoh :

```
10  cls
    Rem kombinasi himpunan R orang dari jumlah N orang
    Input "Jumlah orang :"; N
    Input "Anggota Himpunan :"; R
    Rem menghitung (N-R)
    X = N-R
    GOSUB 20 F1=F
    Rem menghitung R
    X = R : GOSUB 20 F2=F
    Kombinasi = F2/(F1*F)
    Print "Kombinasi = "; kombinasi
    End
20  ' Program bagian menghitung factorial
    F = 1
    For I = 1 to X
        F = F * I
    Next I
    Return
```

Latihan untuk pemahaman

Gunakan statemen subroutine untuk menghitung akar-akar persamaan kuadrat (program VII)

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM X Variabel Berindex

Variable berindex digunakan untuk menulis data dengan variable berulang misalnya, data pengukuran, matriks, data mahasiswa, dll. Untuk menyatakan variable berindex digunakan statemen DIM, misalnya :

- DIM (100)
- DIM A (10,10)

Contoh 1 :

```
Rem menghitung standar deviasi
cls
DIM X (100)
Input "Jumlah data "; N
For I = 1 TO N
    Print "Data ke "; i
    Input X (I)
    Total = total+1
Next I
Rem menghitung rata-rata
Rata = total/n
Rem menghitung sigma (X(i)-rata)
For I = 1 TO N
    S1=S1+(X(I)-rata)/2
Next I
sd = sqrs (S1/n)
Print "standard deviasi = "; sd
End
```

Contoh 2 :

```
Rem mengisi data mahasiswa
Dim M (I), N$(I), C$(I)
cls
Input "banyaknya mahasiswa ="N
For I=1 TO N
Input "nama mahasiswa : "N$(I)
Input "NIM = ";N$( I)
Input "alamat =", C$( i)
Print "nama mahasiswa: ";N$(I); "NIM : ";M(I); "alamat
:";C$(I)
Next i
End
```

Jalankan program diatas dengan mengisi simulasi data mahasiswa yang banyaknya >10 mahasiswa

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM XI MATRIX

Matrix disebut variable berdimensi (berindex) 2, posisi data ditentukan oleh 2 atau lebih ekspresi yaitu baris dan kolom. Untuk bisa memahami operasi matriks, terlebih dahulu harus dipahami operasi kalang dan variable berindeks.

Contoh 1 program menulis matriks nol

```
CLS
Input "Banyak baris "; M
Input "Banyak kolo," " " N
DIM A(M,N)
Rem GOSUB program
GOSUB 20
Print
Print " HASIL MATRIX SEMUA NOL"
PRINT " _ _ _ _ _ _ _ _ _ _"
FOR I = 1 TO m
    FOR J=1 TO N
        PRINT A(I,J)
    NEXT j
    PRINT >>
NEXT I
END ~
20 'membuat matrik nol
    FOR J=1 TO N
        A(I,J)=0
    NEXT j
NEXT i
Return
```

Contoh 2 program operasi penjumlahan matriks

```
CLS
Input "Banyak baris "; M
Input "Banyak kolom, "; N
DIM A(M,N) , B(M, ,N) , C(M,N)
Rem masukkan data matrik A
Print
Print "Data untuk matrik A"
FOR I =1 TO m
    FOR J=1 TO N
        PRINT "A(";I;" , ";J;" ) " ;:Input A(I,J)
    NEXT J
NEXT I
Rem masukkan data matrik B
Print
Print "Data untuk matrik B"
FOR I =1 TO m
    FOR J=1 TO N
        PRINT "A(";I;" , ";J;" ) " ;:Input A(I,J)
    NEXT J
NEXT I
```


Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

PRAKTIKUM XII BERKAS

Ada 2 jenis berkas data, berkas masukan urut dan berkas data masukan acak. Dalam praktikum ini akan dibahas mengenai berkas data masukan urut.

1. Membuka berkas

BU : OPEN namaberkas FOR mode AS [#] nomor

- OPEN, FOR, AS adalah statement Quick Basic
- Nama berkas adalah nama berkas yang dibuka. Folder file juga harus ditulis lengkap
- Mode harus dipilih apakah INPUT (sebagai masukan) atau OUTPUT (sebagai keluaran)

Contoh : OPEN "D:\DATA\latihan1.dat"FOR OUTPUT AS #1

Artinya kita membuka file di direktori D, folder Data dengan nama file latihan1.dat yang digunakan sebagai keluaran

2. Menutup berkas

BU : CLOSE [#nomor]

Berkas yang sudah dipakai harus ditutup kembali dengan tujuan agar data yang tadi dipakai ditulis kembali kedalam berkas sehingga tidak ada data yang hilang dan juga agar data tersebut bias digunakan oleh progam lain.

3. Menulis dan membaca berkas

Untuk menulis berkas digunakan statement WRITE dan untuk membaca berkas digunakan statemen INPUT

BU : WRITE [#nomor], var

INPUT [#nomor], var

Var merupakan daftar variabel untuk menyimpan data atau menerima data yang dibaca

Contoh 1 program menulis berkas

```
CLS
OPEN "contoh.dat" FOR output AS #1
For I = 1 to 5
    Input "Nama Mahasiswa :";nama$
    Input "Alamat Mahasiswa :";alamat$
    Write#1,nama$,alamat$
Next I
Close #1
End
```

Contoh 2 program membaca berkas

```
CLS
OPEN "contoh.dat" FOR output AS #1
For I = 1 to 5
    Input#1,nama$,alamat$
    Print nama$, alamat$
Next I
End
```

Latihan untuk pemahaman

Buatlah sebuah program menghitung jarak optis dari 5 pengukuran (data bebas). Kemudian simpanlah hasilnya dalam berkas bernama optis.dat dalam bentuk table.

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |

Lembar Kerja Mahasiswa

| | | |
|------------------|--|-----------|
| Nama Mahasiswa : | | Tanggal : |
| NRP : | | Asisten |
| Praktikum ke : | | |
| | | |