

# 1 D4 - TEKKOM B

## STATEMENT PENENTU KEPUTUSAN



Nama	:	Septian Bagus Jumanoro
Kelas	:	1B – D4 Teknik Komputer
NRP	:	3221600039
Dosen	:	Ir Sigit Wasista, M.Kom
Mata Kuliah	:	Praktikum Pemrograman Dasar 1
Hari/Tgl. Praktikum	:	17 September 2021

## PERCOBAAN 4

### 3.1 JUDUL: STATEMENT PENENTU KEPUTUSAN

3.2 TUJUAN: Mempelajari dan mengamati penggunaan statement penentu keputusan yaitu statement if-else. Memanfaatkan statement if-else untuk membuat proses perulangan yaitu dengan dikombinasikan dengan statement goto-label. Mempelajari statement if-else bersarang (nested if). Mempelajari bentuk lain dari statement if-else yaitu statement else-if. Mempelajari penggunaan dari statement switc -case sebagai alternatif pengganti dari statement else-if.

3.3 TEORI: Statement if-else digunakan untuk menentukan pilihan dari suatu kondisi yang diberikan. Cara kerjanya adalah melakukan perbandingan menggunakan Relational Operator dan/atau Logical Operator, apabila suatu kondisi terpenuhi (benar) maka akan mengerjakan proses A dan apabila kondisi tidak terpenuhi (salah) maka tidak akan mengerjakan proses A atau dapat juga mengerjakan proses lain misalkan proses B. Proses A dan B dapat berupa satu baris statement atau beberapa baris yang dikelompokkan (Compound Statement). Statement if-else bersarang merupakan kombinasi dari beberapa if-else, dinamakan apabila ada beberapa kondisi yang harus diuji kebenarannya secara bersama (if didalam if). Statement else-if adalah cara lain dari bentuk if-else, yang digunakan untuk menentukan satu kondisi yang benar dari beberapa kondisi yang tersedia. Statement switc -case adalah bentuk lain dari statement else-if, dengan mendaftarkan kondisi secara vertical dalam satu kolom sehingga memudahkan dalam hal evaluasi program.

### 3.4 PROGRAM PERCOBAAN

3.4.1 Mengamati penggunaan statement penentu keputusan sederhana, yaitu menentukan apakah bilangan yang dimasukkan lebih kecil dari nol?

```
/* Nama File : IF-ELSE1.C */

void main()
{
    int bil,absolut;

    printf("Masukkan bilangan bulat: ");
    scanf("%d", &bil);

    if(bil < 0) absolut = -bil;

    printf("Nilai absolute dari %d adalah = %d", bil, absolut);

    getch ();
}
```

**3.4.2** Suatu bilangan apabila dapat dibagi dengan bilangan yang lain, maka tidak menghasilkan suatu sisa nilai, sehingga dengan cara ini program dapat menentukan apakah suatu bilangan habis dibagi atau tidak. Program dibawah ini menghitung sisa pembagian menggunakan operator aritmatika yaitu ‘%’ (modulus).

```
/* Nama File : IF-ELSE2.C */

main()
{
    int pembilang, penyebut, sisa;

    printf("Masukkan pembilang : ");
    scanf("%d", &pembilang);
    printf("Masukkan penyebut : ");
    scanf("%d", &penyebut);

    sisa = pembilang % penyebut;

    puts("");
    if(sisa)
        printf("%d tidak abis dibagi %d\n", pembilang, penyebut);
    else
        printf("%d abis dibagi %d\n", pembilang, penyebut);

    getch ();
}
```

**3.4.3** Menentukan keputusan bahwa suatu nilai termasuk dalam suatu batas jangkauan suatu nilai tertentu. Contoh, semua karakter huruf kecil mempunyai kode ASCII mulai dari huruf a sampai z, yaitu dari 97 sampai 122. Program dibawah ini menunjukkan bahwa suatu karakter yang dimasukkan melalui keyboard adalah termasuk karekter huruf kecil atau bukan.

```
/* Nama File : IF-ELSE3.C */

void main()
{
    c ar c;

    printf("Masukkan suatu karakter : "); scanf("%c", &c);

    if(cS='a' && cc='z')
        printf("\nKarakter tersebut termasuk uruf kecil.\n");
    else
        printf("\nKarakter tersebut bukan uruf kecil.\n");

    getch ();
}
```

**3.4.4** Suatu proses ada kalanya dilakukan berulang, untuk mendapatkan nilai perhitungan yang lebih teliti (iterasi) atau untuk mengolah suatu data nilai secara tabel seperti pada daftar nilai siswa. Berikut diberikan contoh sederhana, yaitu menampilkan penambahan suatu nilai index dimana sering digunakan untuk proses perulangan. Dengan menggunakan kombinasi if-goto maka didapatkan satu statement perulangan.

```
/* Nama File : IF-ELSE4.C */

void main()
{
    int i=0, asil=1;

    ulang:

        i++;
        asil*=2;

        printf("Iterasi ke %2d: %5d\n", i, asil);

        if(i<10) goto ulang;

        getch ();
}
```

**3.4.5** Compound Statement adalah mengelompokkan suatu proses menjadi satu blok. Didalam if-else penggunaan Compound Statement adalah untuk mengelompokkan proses-proses yang harus dikerjakan bila kondisi terpenuhi ataupun bila kondisi tidak terpenuhi.

```
/* Nama File : IF-ELSE5.C */

void main()
{
    char tag;

    printf("Masukkan tipe tag ( 2 / 4 ) : "); scanf("%c", &tag);

    if(tag=='2') {
        printf("Merk sepeda motor bertipe 2 tag\n\n");
        printf("    1. YA MAHA");
        printf("    2. YA SUZUKI");
        printf("    3. YA HONDA");
    } else {
        printf("Merk sepeda motor bertipe 4 tag\n\n");
        printf("    1. HONDA");
        printf("    2. SUZUKI");
        printf("    3. YAMAHA");
    }

    getch ();
}
```

3.4.6 Statement `if-else` tidak dapat digunakan untuk menentukan satu pilihan yang benar diantara beberapa pilihan yang ada, oleh karena itu dapat digunakan statement `else-if` dimana dapat digunakan untuk menguji kondisi yang benar diantara beberapa beberapa kondisi yang diberikan. Program dibawah ini adalah digunakan untuk menentukan satu pilihan diantara beberapa pilihan yaitu memilih satu operator (dimasukkan dari keyboard) yang sesuai diantara 4 Arithmetic Operator yaitu (+, -, / dan \*). Bila suatu operator dikenal maka dilanjutkan proses perhitungan sesuai dengan operator yang dikenal tersebut, sedangkan bila tidak dikenal maka diberikan komentar "Operator SALAH!". Pengujian operator dilakukan secara bertahap dari operator '\*', '/', '+' dan '-', bila keempat operator tidak ditemukan maka pada statement penujian cukup dituliskan `else` tanpa `if`, yang menandakan pengujian selesai dan tidak ditemukan operator yang sesuai.

```
/* Nama File : IF-ELSE6.C */

void main()
{
    int  valid_operator = 1;
    char operator;
    float bil1, bil2,  asil;

    printf("Masukkan 2 bilangan dan sebua operator.\n\n");
    printf("Dengan format:\n\n");
    printf("= bilangan-1 operator bilangan-2\n\n");
    printf("= ");

    scanf("%f %c %f", &bil1, &operator, &bil2);

    if(operator == '*') {
        asil = bil1 * bil2;
    } else if(operator == '/')
        asil = bil1 / bil2;
    else if(operator == '+')
        asil = bil1 + bil2;
    else if(operator == '-')
        asil = bil1 - bil2;
    else
        valid_operator = 0;

    if(valid_operator) {
        printf("\nHasil per itungan\n\n");
        printf("%g %c %g = %g\n",bil1,operator,bil2, asil);
    } else {
        printf("\nOperator SALAH!\n");
        printf("\nGunakan operator +, -, / dan * saja!");
    }

    getch ();
}
```

3.4.7 Untuk lebih menstrukturkan susunan program, maka penggunaan dari else-if dapat diganti dengan statement `switch -case`. Program dibawah ini adalah modifikasi dari program diatas. Statement `switch -case` akan membuat program terlihat lebih terstruktur, dimana statement `else` pada pengujian diatas diganti dengan statement `default`. Setiap blok `case` harus diakhiri dengan statement `break`, bila diinginkan hanya satu kondisi yang memenuhi, dan bila statement `break` tidak digunakan maka beberapa kondisi tersebut akan diuji secara bersama, sehingga kemungkinan ada dua atau beberapa kondisi yang benar.

```
/* Nama File : IF-ELSE7.C */

void main()
{
    int  valid_operator = 1;
    char operator;
    float bil1, bil2,  asil;

    printf("Masukkan 2 bilangan dan sebua operator.\n\n");
    printf("Dengan format:\n\n");
    printf("= bilangan-1 operator bilangan-2\n\n");
    printf("= ");

    scanf("%f %c %f", &bil1, &operator, &bil2);

    switch (operator) {
        case '*': asil = bil1 * bil2; break;
        case '/': asil = bil1 / bil2; break;
        case '+': asil = bil1 + bil2; break;
        case '-': asil = bil1 - bil2;
        default : valid_operator = 0;
    }
    if(valid_operator) {
        printf("\nHasil per itungan\n\n");
        printf("%g %c %g = %g\n",bil1,operator,bil2, asil);
    } else {
        printf("\nOperator SALAH!\n");
        printf("\nGunakan operator +, -, / dan * saja!");
    }

    getch ();
}
```

### 3.5 TUGAS-TUGAS (Dikumpulkan 1 minggu setelah praktikum dilaksanakan)

3.5.1 Buatlah program untuk menentukan bilangan ganjil atau bilangan genap dari bilangan yang dimasukkan melalui keyboard.

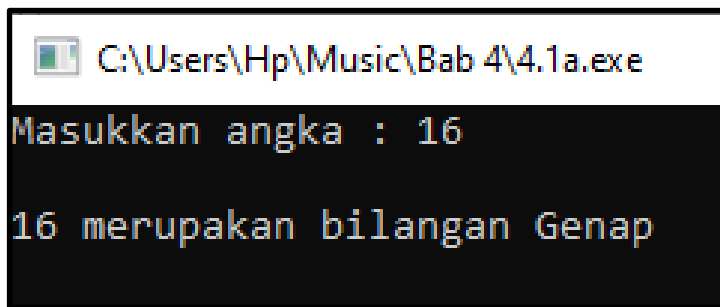
- **Source Code**

```
#include <stdio.h>
void main()
{
    int angka;
    printf("Masukkan angka : ");
    scanf("%d", &angka);

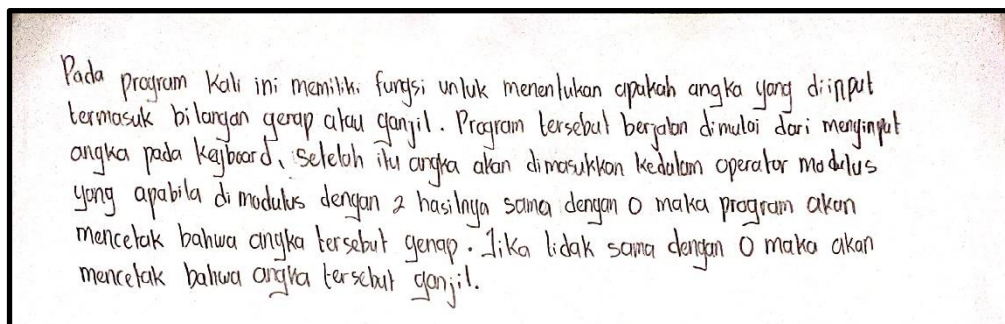
    if (angka % 2 == 0)
        printf("\n%d merupakan bilangan Genap", angka);
    else
        printf("\n%d merupakan bilangan Ganjil", angka);

    getch();
}
```

- **Output**



- **Analisa**



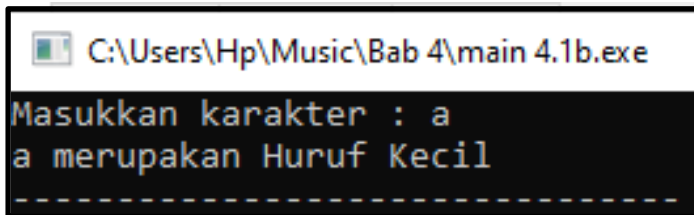
3.5.2 Buatlah program untuk menentukan kelompok suatu karakter yang dimasukkan melalui keyboard. Kelompok karakter tersebut adalah huruf kecil, huruf besar, angka, dan karakter khusus ( tanda baca, operator dsb).

- **Source Code**

```
#include <stdio.h>
void main()
{
    char karakter;
    printf("Masukkan karakter : ");
    scanf("%c", &karakter);

    if (karakter >= 'a' && karakter <= 'z')
        printf("%c merupakan Huruf Kecil", karakter);
    else if (karakter >= 'A' && karakter <= 'Z')
        printf("%c merupakan Huruf Kapital", karakter);
    else if (karakter >= '0' && karakter <= '9')
        printf("%c merupakan Angka", karakter);
    else
        printf("%c merupakan Karakter Khusus", karakter);
}
```

- **Output**



- **Analisa**

Pada program kali ini memiliki fungsi untuk menentukan apakah karakter yang diinput termasuk huruf kecil atau kapital, angka atau karakter khusus. Dikarenakan karakter maka tipe variabelnya char dan syntax memanggilnya %c. Pada saat karakter diinput sesuai dengan salah satu statement maka akan tercetak sesuai dengan statementnya. Program ini menggunakan persamaan kode ASCII yang membandingkannya dengan input.

### 3.5.3 Buatlah program kalkulator sederhana. Program harus memungkinkan user untuk mengetik ekspresi dengan bentuk:

Bilangan Operator

- Operator yang digunakan adalah: \* / + - % & | S E
- Operator s memerintah program untuk men-set "accumulator" untuk menyimpan angka yang diketik oleh user.
- Operator E memerintah program untuk mengakhiri program.



### Contoh running program:

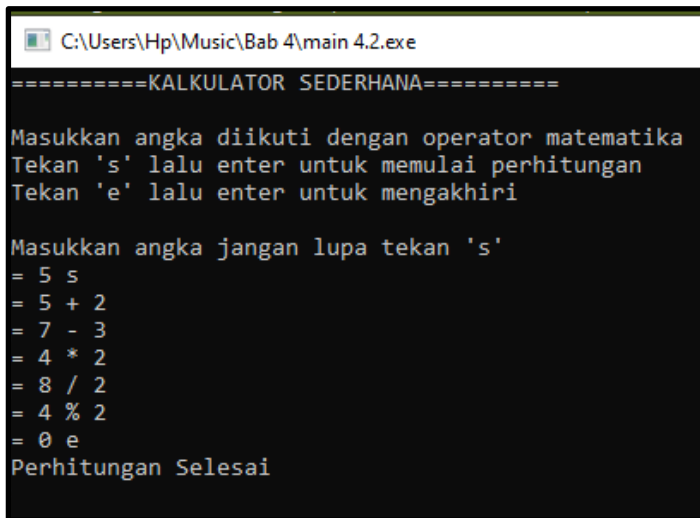
Mulai per itungan	
255 S	= > men-set accumulator
= 255.000000	
10 /	=> membagi accumulator dengan 10
= 25.500000	
15 -	=> mengurangi accumulator dengan 15
= 10.500000	
0 E	=> selesai
Ak ir per itungan	

- **Source Code**

```
#include <stdio.h>
#include <math.h>
void main()
{
    float a, b, hasil;
    char oper;
    printf("=====KALKULATOR SEDERHANA=====\\n\\n");
    printf("Masukkan angka diikuti dengan operator matematika\\n");
    printf("Tekan 's' lalu enter untuk memulai perhitungan\\n");
    printf("Tekan 'e' lalu enter untuk mengakhiri\\n\\n");
    printf("Masukkan angka jangan lupa tekan 's'\\n");
    printf("= ");
    scanf("%f %c", &a, &oper);
    if(oper == 's')
    {
        printf("= %g ", a);
        ulang:
        oper = getche();
        if (oper == 'e')
        {
            printf("\\nPerhitungan Selesai\\n");
        }
        else if(oper)
        {
            scanf("%f", &b);
            if (oper == '+') {hasil = a + b;}
            else if (oper == '-') {hasil = a - b;}
            else if (oper == '*') {hasil = a * b;}
            else if (oper == '/') {hasil = a / b;}
            else if (oper == '%') {hasil = (int)a % (int)b;}
            else if (oper == '&') {hasil = (int)a && (int)b;}
            else if (oper == '|') {hasil = (int)a || (int)b;}
            else {printf("Perhitungan Selesai");}
            printf("= %g ", hasil);
            a = hasil;
            goto ulang;
        }
    }
    else {printf("Harap masukkan 's' setelah angka");}

    getche();
}
```

- **Output**



```
C:\Users\Hp\Music\Bab 4\main 4.2.exe
====KALKULATOR SEDERHANA====
Masukkan angka diikuti dengan operator matematika
Tekan 's' lalu enter untuk memulai perhitungan
Tekan 'e' lalu enter untuk mengakhiri

Masukkan angka jangan lupa tekan 's'
= 5 s
= 5 + 2
= 7 - 3
= 4 * 2
= 8 / 2
= 4 % 2
= 0 e
Perhitungan Selesai
```

- **Analisa**

Program kali ini merupakan kalkulator sederhana. Awal jalannya program saat user menginputkan angka dan menekan "s" maka program akan start. Lalu user memasukkan operator bilangan. Jika setelah itu user tidak mengklik huruf "e" maka program akan terus berjalan karena terdapat looping di dalam source codenya. Jika user mengklik "e" maka program akan berhenti.

3.5.4 Buatlah program untuk menentukan suatu tahun kabisat atau bukan yang dimasukkan melalui keyboard, dimana tahun dibatasi mulai dari tahun 1900 sampai dengan tahun 2005.

- **Source Code**

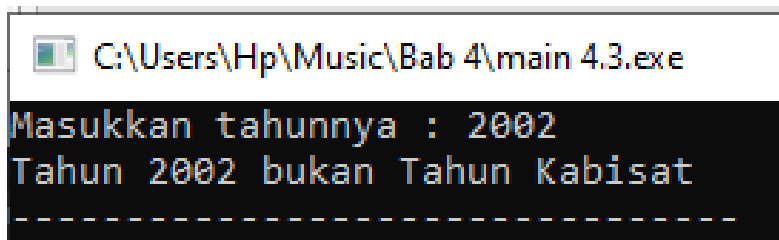
```
#include <stdio.h>
void main()
{
    int year, range;
    printf("Masukkan tahunnya : ");
    scanf("%d", &year);
    range = year;
    if (range >= 1900 && range <= 2005)
    {
        if (year % 400 == 0)
        {
            printf("Tahun %d merupakan Tahun Kabisat", year);
        }
        else if (year % 400 != 0 && year % 100 == 0)
        {
            printf("Tahun %d bukan Tahun Kabisat", year);
        }
        else if (year % 400 != 0 && year % 100 != 0 && year % 4 == 0)
        {
            printf("Tahun %d merupakan Tahun Kabisat", year);
        }
    }
}
```

```

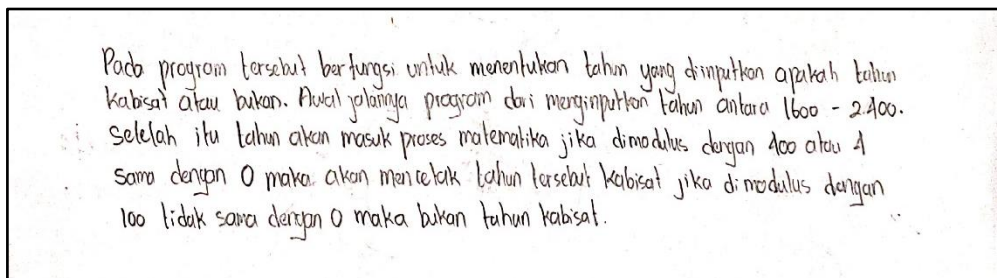
        else if (year % 400 != 0 && year % 100 != 0 && year % 4 != 0)
        {
            printf("Tahun %d bukan Tahun Kabisat", year);
        }
    }
    else
    {
        printf("Masukkan Tahun yang sesuai antara 1600 - 2400");
    }
}

```

- **Output**



- **Analisa**



**3.5.5** Buatlah program untuk menghitung determinan (D) dan mencari akar-akar dari persamaan kuadrat :  $Ax^2 + x + = 0$  dimana  $D = B^2 - 4AC$  .

- Jika  $D = 0$  , maka akar-akarnya kembar, yaitu:  $x_1 = x_2 = \frac{-B}{2A}$

- Jika  $D > 0$  , maka akar-akarnya berbeda, yaitu:

$$x_1 = \frac{(-B + \sqrt{D})}{2A} \text{ dan } x_2 = \frac{(-B - \sqrt{D})}{2A}$$

- Jika  $D < 0$  , maka akar-akarnya imajiner berlainan, yaitu:

$$x_1 = \frac{(-B + \sqrt{D})}{2A} j \text{ dan } x_2 = \frac{(-B - \sqrt{D})}{2A} j$$

- **Source Code**

```
#include <stdio.h>
#include <math.h>

void main()
{
    float A, B, C, D, x1, x2;
    printf("PROGRAM MENGHITUNG DETERMINAN (D)\n");
    printf("Masukkan nilai A: ");
    scanf("%f", &A);
    printf("Masukkan nilai B: ");
    scanf("%f", &B);
    printf("Masukkan nilai C: ");
    scanf("%f", &C);

    D = B * B - (4 * A * C);
    printf("Determinannya: %g", D);

    if (D == 0)
    {
        x2 = x1;
        x1 = -B / (2 * A);
        printf("\nPersamaan tersebut memiliki akar-akar yang kembar\n");
        printf("Maka nilai x1 = %g", x1);
        printf("\nMaka nilai x2 = %g", x2);
    }
    else if (D > 0)
    {
        x1 = ((-B + sqrt(D)) / (2 * A));
        x2 = ((-B - sqrt(D)) / (2 * A));
        printf("\nPersamaan tersebut memiliki akar-akar yang
berbeda\n");
        printf("Maka nilai x1 = %g", x1);
        printf("\nMaka nilai x2 = %g", x2);
    }
    else if (D < 0)
    {
        x1 = ((-B + sqrt(-D)) / (2 * A));
        x2 = ((-B - sqrt(-D)) / (2 * A));
        printf("\nPersamaan tersebut memiliki akar-akar yang
imajiner\n");
        printf("Maka nilai x1 = %g * j", x1);
        printf("\nMaka nilai x2 = %g * j", x2);
    }
}
```

- **Output**

```
C:\Users\Hp\Music\Bab 4\main 4.4.exe
PROGRAM MENGHITUNG DETERMINAN (D)
Masukkan nilai A: 2
Masukkan nilai B: 4
Masukkan nilai C: 6
Determinannya: -32
Persmaan tersebut memiliki akar-akar yang imajiner
Maka nilai x1 = 0.414214 * j
Maka nilai x2 = -2.41421 * j
-----
```

- **Analisa**

Program kali ini berfungsi untuk menentukan persamaan suatu akar-akar. Dimana user memasukkan input untuk variabel A, B, C dan nanti akan masuk pada operasi matematika. Untuk mencari determinan menggunakan  $B^2 - 4AC$ . Pada program untuk operasi perakaran menggunakan "sqrt" diikuti dengan determinan menjadi  $\sqrt{D}$ . Jika hasil determinan sama dengan 0 maka persamaan tersebut akarnya kembar. Jika determinan  $\geq 0$  maka persamaan tersebut akarnya berbeda. Jika determinan  $< 0$  maka persamaan tersebut akar-akarnya imajiner.