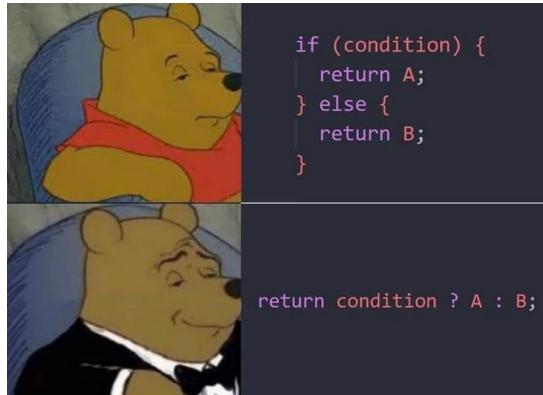


## Modul 4

### Pemilihan



#### A. Tujuan

1. Praktikan dapat memahami dan mengimplementasikan konsep pemilihan/percabangan dalam Bahasa Pemrograman C
2. Praktikan dapat mengetahui waktu yang tepat dalam penggunaan pemilihan/percabangan pada studi kasus tertentu

#### B. Penjelasan

Dalam kehidupan sehari-hari, kita selalu dihadapkan pada beberapa pilihan atau diharuskan memilih salah satu dari opsi yang ada dimana pilihan-pilihan tersebut didasarkan atas kondisi tertentu. Namun, pilihan tersebut hanya dapat dilakukan apabila kondisi telah terpenuhi. Hal ini juga memiliki kesamaan pada materi **pemilihan/percabangan** dalam bahasa pemrograman.

Pemilihan atau percabangan merupakan sebuah algoritma kondisi yang mana terdapat persyaratan tertentu yang akan dieksekusi dan memenuhi suatu kondisi. Pada saat melakukan pembuatan program yang lebih kompleks, pemilihan/percabangan akan selalu dijumpai untuk menyelesaikan suatu permasalahan kode pemrograman. Pada pemilihan sendiri memiliki 2 nilai yaitu **true** dan **false**, suatu aksi hanya akan dilakukan jika nilai dari kondisinya bernilai benar atau salah. Kedua nilai ini selalu mempengaruhi bagaimana nantikan program yang dibuat akan bekerja.

- **OPERATOR BOOLEAN**

Operator Boolean adalah operator yang digunakan untuk membandingkan suatu kondisi/persyaratan didalam pengkondisian, hasil yang didapatkan dari operator ini bernilai true atau false. Bentuk penggunaan dari operator Boolean yaitu,

ruas 1 <operator> ruas 2

Operator	Keterangan	Contoh
<code>==</code> (sama dengan)	BerNilai True jika ruas kiri dan kanan memiliki nilai yang sama	$a = 1, b = 0$  operasi : $a == b$ (FALSE)
<code>!=</code> (tidak sama dengan)	BerNilai True jika nilai dari ruas kiri dan kanan berbeda	$a = 2, b = 1$  operasi : $a != b$ (TRUE)
<code>&lt;</code> (kecil dari)	BerNilai True jika nilai dari ruas kiri lebih kecil dari ruas kanan	$a = 7, b = 3$  operasi : $a < b$ (FALSE)
<code>&gt;</code> (besar dari)	BerNilai True jika nilai dari ruas kiri lebih besar dari ruas kanan	$a = 10, b = 5$  operasi : $a > b$ (TRUE)
<code>&lt;=</code> (kecil dari sama dengan)	BerNilai True jika nilai dari ruas kiri lebih kecil atau sama dengan ruas kanan	$a = 4, b = 9, c = 10$  operasi : $a <= b$ (TRUE) $c <= b$ (FALSE)
<code>&gt;=</code> (besar dari sama dengan)	BerNilai True jika nilai dari ruas kiri lebih besar atau sama dengan ruas kanan	$a = 3, b = 10, c = 2$  operasi : $a >= b$ (FALSE) $b >= c$ (TRUE)
<code>!</code> (negasi)	BerNilai True jika nilai komplemen atau kebalikan dari suatu kondisi	$a = 2, b = 5, c = 4$  operasi : $!(b == a)$ (TRUE) $!(c < b)$ (FALSE)
<code>&amp;&amp;</code> (dan)	BerNilai True jika nilai di ruas kiri dan kanan keduanya bernilai True	$a = 5, b = 2, c = 7$  operasi : $a > b \&\& b < c$ (FALSE) $b <= c \&\& c > a$ (TRUE)
<code>  </code> (atau)	BerNilai True jika nilai di ruas kiri dan kanan salah satunya bernilai True	$a = 1, b = 1, c = 5$  operasi : $a == b    a > c$ (TRUE) $a > c    a != b$ (FALSE)

- **PEMILIHAN**

Pada pemilihan/percabangan terdapat 3 jenis umum pemilihan yaitu IF-ELSE, SWITCH-CASE dan OPERATOR TERNARY. Berikut merupakan contoh dari 3 pemilihan tersebut :

### A. IF-ELSE

#### 1) Satu Kasus

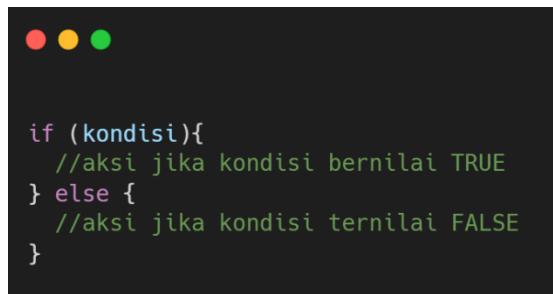


```
if (kondisi){  
    //aksi jika kondisi bernilai TRUE  
}
```

Gambar 1.A Satu Kasus (If)

Aksi akan dieksekusi jika kondisi tersebut terpenuhi, namun jika kondisi tidak terpenuhi maka aksi tidak akan tereksekusi

#### 2) Dua Kasus

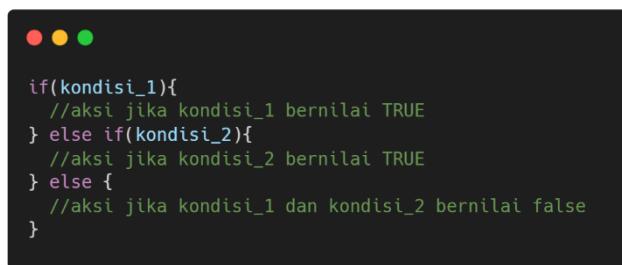


```
if (kondisi){  
    //aksi jika kondisi bernilai TRUE  
} else {  
    //aksi jika kondisi ternilai FALSE  
}
```

Gambar 2.A Dua Kasus (If-Else)

Aksi pada bagian if akan dijalankan jika kondisi tersebut memenuhi syarat, dan jika kondisi tidak memenuhi syarat maka aksi pada bagian else akan dijalankan

#### 3) Tiga atau lebih banyak kasus



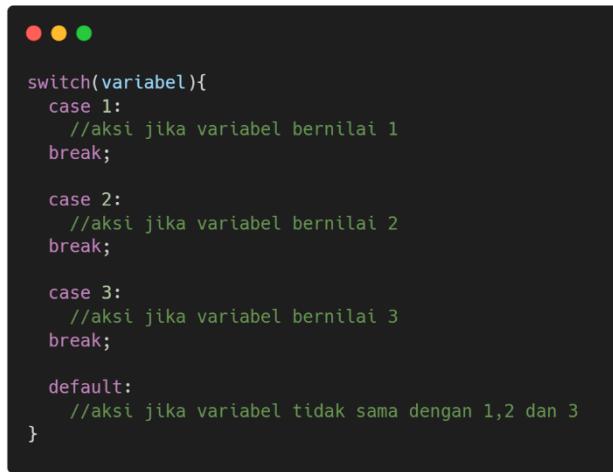
```
if(kondisi_1){  
    //aksi jika kondisi_1 bernilai TRUE  
} else if(kondisi_2){  
    //aksi jika kondisi_2 bernilai TRUE  
} else {  
    //aksi jika kondisi_1 dan kondisi_2 bernilai false  
}
```

Gambar 3.A Lebih dari 2 Kasus (If-Else if - Else)

Aksi akan dijalankan jika kondisi 1 terpenuhi, jika kondisi 1 tidak terpenuhi maka program akan memeriksa pada kondisi 2, dan jika kondisi 1 dan kondisi 2 tidak terpenuhi maka program akan langsung mengeksekusi aksi pada bagian else.

## B. SWITCH-CASE

Jenis pemilihan kedua yaitu switch-case yang biasanya digunakan untuk kasus pemilihan yang lebih banyak, dan juga supaya baris kode yang dibuat sederhana dan rapi. Switch-case ini sering juga digunakan untuk membuat menu pada program yang menggunakan inputan integer maupun character.



```
switch(variabel){
    case 1:
        //aksi jika variabel bernilai 1
        break;

    case 2:
        //aksi jika variabel bernilai 2
        break;

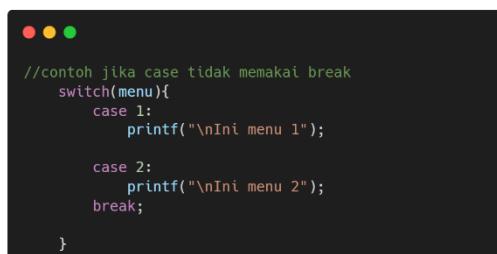
    case 3:
        //aksi jika variabel bernilai 3
        break;

    default:
        //aksi jika variabel tidak sama dengan 1,2 dan 3
}
```

Gambar 4.B Penggunaan Switch case

Pada switch, kondisi dari sebuah percabangan terletak setelah ‘case’. Pada gambar diatas, aksi akan dijalankan jika nilai dari sebuah variabel menunjukkan angka 1, 2 atau 3 dan jika ketiga nilai itu tidak terpenuhi maka aksi **default** akan dijalankan, default berarti else. **Break** pada pemilihan switch berfungsi untuk memaksa program keluar dari suatu kondisi pada pemilihan agar aksi pada kondisi selanjutnya tidak dijalankan. Jika pernyataan **break** tidak disertakan, program akan terus berjalan hingga menemukan perintah **break** atau mencapai kondisi penghentian lainnya. Berikut adalah contoh penggunaan break:

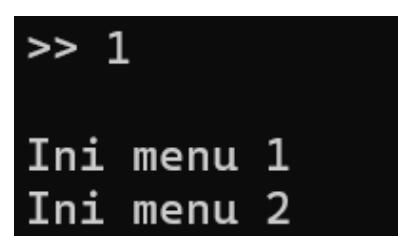
### 1) Jika tidak menggunakan break



```
//contoh jika case tidak memakai break
switch(menu){
    case 1:
        printf("\nIni menu 1");

    case 2:
        printf("\nIni menu 2");
        break;
}
```

Gambar 5.B Switch-case tidak menggunakan break



```
>> 1
Ini menu 1
Ini menu 2
```

Gambar 6.B Hasil Output

Berdasarkan gambar diatas, jika kalian mengakses *case 1* program akan terus berjalan hingga menemukan perintah **break** sehingga *output* yang dihasilkan seperti pada **gambar 6.B**.

## 2) Jika menggunakan break

```
● ● ●

//contoh jika case memakai break
switch(menu){
    case 1:
        printf("\nIni menu 1");
        break;

    case 2:
        printf("\nIni menu 2");
        break;
}
```

```
>> 1
Ini menu 1
```

Gambar 7.B Switch-case menggunakan break

Gambar 8.B Hasil Output

Berdasarkan gambar diatas, jika kita ingin mengakses *case 1* program akan menampilkan *output* seperti pada **Gambar 8.B** sesuai dengan kode program yang dibuat. Sama halnya jika kita ingin mengakses *case 2*, program akan langsung menampilkan *output* “Ini menu 2”.

## C. OPERATOR TERNARY

Operator ternary atau Conditional Operator adalah operator yang digunakan untuk membuat Keputusan dalam satu baris kode. Operator ini memiliki tiga operand dan digunakan untuk melakukan evaluasi suatu kondisi, kemudian menghasilkan nilai berdasarkan kondisi tersebut.

```
● ● ●

kondisi ? variabel1 : variabel2
```

Gambar 9.C Rumus operator ternary

Jika kondisi bernilai true, maka variabel1 akan dihasilkan, sedangkan jika kondisi bernilai false, maka variabel2 akan dihasilkan.

```
● ● ●

//Contoh :
int a = 5;
int b = 10;
int max = (a<b) ? <aksi1> : <aksi2>;
```

Gambar 10.C Contoh Penggunaan Operator Ternary

- **STRING**



```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

typedef char string[50];
```

Gambar 11. library <string.h> dan tipe data buatan string

Pada Bahasa pemrograman C tipe data string merupakan tipe data bentukan, dimana dibentuk dari tipe data char dengan mendeklarasikan `typedef char string[<max karakter>]` di header. Pada pemilihan akan erat kaitannya dengan suatu pengkondisian bertipe data string, namun untuk bisa menggunakan fungsi-fungsi yang akan membantu dalam perbandingan didalam pengkondisian ini perlu memanggil library `#include <string.h>` pada header. Didalam library ini terdapat beberapa fungsi yaitu strcpy, strlen, strcmp dan strcmpi. Berikut adalah penjelasan lebih dalam :

### 1. STRCPY

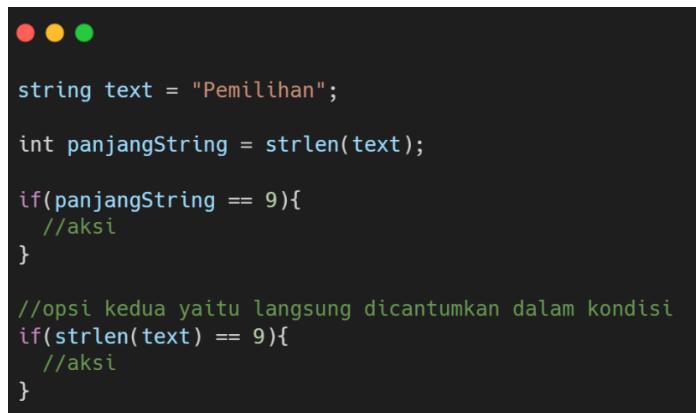


```
strcpy(variabelA, variabelB);
strcpy(variabelA, "Pemilihan");
```

Gambar 12.1 Contoh Penggunaan strcpy

Fungsi strcpy merupakan *string copy*, yang berfungsi untuk menyalin sebuah *string* kedalam suatu variabel yang terletak diruas kirinya. Seperti contoh di atas menyalin nilai dari variabelB kedalam variabelA dan memasukan string “Pemilihan” kedalam variabelA.

## 2. STRLEN



```
string text = "Pemilihan";
int panjangString = strlen(text);

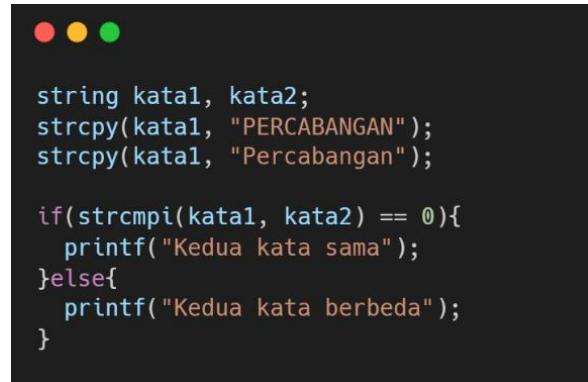
if(panjangString == 9){
    //aksi
}

//opsi kedua yaitu langsung dicantumkan dalam kondisi
if(strlen(text) == 9){
    //aksi
}
```

Gambar 13.2 Contoh Penggunaan `strlen`

Fungsi `strlen` merupakan *string length*, yang berfungsi untuk mengetahui jumlah karakter yang terdapat dalam sebuah *string* atau variabel dengan tipe data *string*. Penggunaan fungsi `strlen` ini akan jauh lebih efektif jika menempel pada suatu variabel / dijadikan sebagai kondisi pada pemilihan.

## 3. STRCMP & STRCMPI



```
string kata1, kata2;
strcpy(kata1, "PERCABANGAN");
strcpy(kata1, "Percabangan");

if(strcasecmp(kata1, kata2) == 0){
    printf("Kedua kata sama");
} else{
    printf("Kedua kata berbeda");
}
```

Gambar 14.3 Contoh Penggunaan `strcasecmp` dan `strcpy`

Fungsi `strcmp` dan `strcasecmp` memiliki manfaat penggunaan yang sama, dimana kedua fungsi ini berfungsi untuk membandingkan dua buah *string*. `strcasecmp` merupakan *string compare insensitive* dimana fungsi ini akan menganggap kedua *string* sama jika semua karakter di variabel `kata1` dan `kata2` identik, tidak peduli dengan huruf kapital atau tidak. Sedangkan `strcmp` merupakan *string compare* dengan *case sensitive*, artinya fungsi akan menganggap kedua string sama / *true* jika semua karakter benar-benar identik termasuk memperhatikan huruf kapitalnya. Kedua fungsi ini sering digunakan sebagai kondisi di pemilihan. Seperti contoh diatas akan mengeluarkan output “Kedua kata sama” karena menggunakan *case*

*insensitive* yaitu strcmpi. Dan jika menggunakan strcmp maka program akan mengeluarkan output kedua kata berbeda.



```
● ● ●
if(strcmp(variabelA, variabelPembanding) == 0)
```

**Gambar 15.3 Contoh Penggunaan strcmp sama dengan**

**Gambar 15.3** akan menghasilkan nilai *True* jika variabelA dan variabelPembanding memiliki isi *string* yang sama.



```
● ● ●
if(strcmp(variabelA, variabelPembanding) != 0)
```

**Gambar 16.3 Contoh Penggunaan strcmp tidak sama dengan**

**Gambar 16.3** akan menghasilkan nilai *True* jika variabelA dan variabelPembanding memiliki isi string yang berbeda.

## GUIDED

1. Buatlah sebuah program yang terdapat 2 menu dalam akun pasien di Rumah Sakit Atma Jaya. Kedua menu tersebut yaitu menu input data pasien dan klasifikasi BMI. Sebelum program masuk pada menu utama, user diharuskan untuk melakukan login terlebih dahulu dengan memasukkan username dan password yang telah dibuat. Setelah berhasil login, user akan masuk ke dalam menu utama. Berikut adalah rincian menu-menu tersebut :

- **Menu Input Data Pasien**

Menu ini berfungsi untuk menerima dan menyimpan sebuah data pribadi user. User diminta untuk memasukkan beberapa data pribadinya seperti nama lengkap, umur dan jenis kelamin. Setelah memasukkan inputan tersebut, program akan menampilkannya dan menambahkan keterangan kategori yang sudah ditentukan dari umur

Umur	Kategori
0 - 5	Balita
6 - 12	Anak-anak
13 - 17	Remaja
18 - 59	Dewasa
>59	Lansia

- **Menu Klasifikasi BMI**

Menu ini berfungsi untuk menentukan massa tubuh user (BMI) dengan menerima inputan terkait dengan berat badan dan tinggi badan user. Dari kedua inputan tersebut, program akan melakukan perhitungan sesuai dengan rumus yang ada. Jika perhitungan sudah selesai, program akan menentukan kategori index massa tubuh

Index	Kategori
< 18.5	Kurus
18.5 – 24.9	Normal
25 – 29.9	Overweight
>29.9	Obesitas

**Note :**

- a. Comment tidak perlu di ketikkan juga

```

● ● ●

#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

#include <string.h>
//library untuk menggunakan fungsi-fungsi string

typedef char string[50];
//membuat tipe data bentukan baru yang bernama string dari kumpulan 50 char

int main(int argc, char *argv[]){
    int menu, umur;
    string nama, jenisKelamin, username;
    string password, kategori, kategoriIndex;
    float berat, tinggi, indexMassaTubuh;

    printf("\t==[LOGIN]==\n");
    printf("\nUsername : ");fflush(stdin);gets(username); //kode untuk memasukkan inputan username
    printf("\nPassword : ");fflush(stdin);gets(password); //kode untuk memasukkan inputan password

    //kondisi untuk mengecek inputan username dan password
    if(strcmp(username, "admin") == 0 && strcmp(password, "admin123") == 0){
        system("cls"); //membersihkan semua tampilan sebelum kode ini

        printf("\t==[ RUMAH SAKIT ATMA JAYA ]==\n");
        printf("\n[1] Input Data Pasien");
        printf("\n[2] Klasifikasi BMI");
        printf("\n> ");scanf("%d", &menu);//inputan menu

        switch(menu){
            case 1:
                printf("\nMasukkan nama lengkap : ");fflush(stdin);gets(nama);
                printf("\nMasukkan umur : ");scanf("%d", &umur);
                printf("\nMasukkan jenis kelamin : ");fflush(stdin);gets(jenisKelamin);

                //kondisi untuk mengecek umur sesuai dengan ketentuan kategori
                if(umur>0 && umur<=5){
                    strcpy(kategori, "balita");
                }else if(umur>6 && umur<=12){
                    strcpy(kategori, "anak-anak");
                }else if(umur>=13 && umur<=17){
                    strcpy(kategori, "remaja");
                }else if(umur>=18 && umur<=59){
                    strcpy(kategori, "dewasa");
                }else{
                    strcpy(kategori, "lansia");
                }

                //menampilkan semua data yang telah diinputkan
                printf("\n\n\t== Data Pasien ==\n");
                printf("\nNama : %s", nama);
                printf("\nUmur : %d", umur);
                printf("\nJenis Kelamin : %s", jenisKelamin);
                printf("\nKategori : %s", kategori);

                //Contoh : Chaterina Olivia | 230712313 | A
                printf("\n\nNAMA PRAKTIKAN | NPM | KELAS");
                break;

            case 2:
                printf("\nMasukkan berat badan Anda (kg) : ");scanf("%f", &berat);
                printf("\nMasukkan tinggi badan Anda (m) : ");scanf("%f", &tinggi);

                //perhitungan memakai rumus index massa tubuh
                indexMassaTubuh = berat / (tinggi * tinggi);

                //kondisi untuk mengecek massa tubuh sesuai dengan ketentuan kategori
                if(indexMassaTubuh < 18.5){
                    strcpy(kategoriIndex, "Kurus");
                }else if(indexMassaTubuh >= 18.5 && indexMassaTubuh <= 24.9){
                    strcpy(kategoriIndex, "Normal");
                }else if(indexMassaTubuh >= 25 && indexMassaTubuh <= 29.9){
                    strcpy(kategoriIndex, "Overweight");
                }else{
                    strcpy(kategoriIndex, "Obesitas");
                }

                //menampilkan output yang sudah ditentukan
                printf("\n\t== Hasil Perhitungan BMI ==\n");
                printf("\nBMI Anda : %.2f", indexMassaTubuh);
                printf("\nKategori : %s", kategoriIndex);

                //Contoh : Chaterina Olivia | 230712313 | A
                printf("\n\nNAMA PRAKTIKAN | NPM | KELAS");
                break;
        }
    }
}

```

```
        default:
            printf("\n[!] Menu tidak tersedia [!]\n");

            //Contoh : Chaterina Olivia | 230712313 | A
            printf("\n\nNAMA PRAKTIKAN | NPM | KELAS");
            break;
    }

}else{
    printf("\n[!] Username & Password Salah [!]\n\n");
    //Contoh : Chaterina Olivia | 230712313 | A
    printf("NAMA PRAKTIKAN | NPM | KELAS\n");
}

return 0;
}
```

**Gambar 17. Jawaban Guided**

 **Hint**

- Praktikan diharapkan untuk melakukan pendalaman materi terkait dengan strcmpi, strcmp, strlen dan operator ternary

**Ketentuan Pengerjaan:**

Pastikan ekstensi program file main adalah .c bukan .cpp

1. Pastikan semua file dikerjakan dalam satu folder (termasuk file .dev dan .c).
2. Guided dikerjakan dalam folder dengan format GD4\_X\_YYYYY lalu di zip.

X = Kelas

YYYYY = 5 digit terakhir NPM

**Kesalahan pengumpulan format penamaan akan dikurangi 10 poin.**

**Harap diperhatikan!**