

MODUL 2: CONTROL STATEMENT: SELECTION STRUCTURE

LAB 2.1: TUJUAN PRAKTIKUM

Learning outcome untuk praktikum pemrograman dasar pada modul 2 ini adalah:

LO 2. Mahasiswa mampu mengimplementasikan dan menganalisis konsep dari pernyataan kontrol pemrograman.

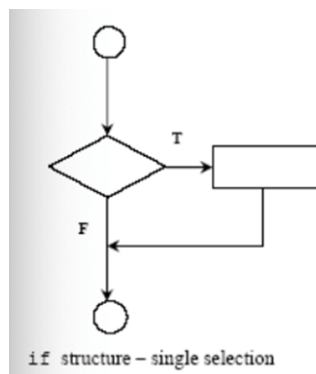
Secara khusus, tujuan dari praktikum pada modul 2 ini adalah:

Mengenal salah satu pengendalian program, yaitu seleksi (*selection structure*) dan mampu mengimplementasikannya dalam program.

LAB 2.2: DASAR TEORI

Eksekusi program akan diawali pada fungsi `main()` dengan proses eksekusi kode secara urut baris demi baris (*line by line*) dan dari atas ke bawah (*top down style*). Namun, pada kenyataannya urutan eksekusi pada *body* fungsi `main()` sangat mungkin tidak urut dari atas ke bawah, namun memerlukan percabangan, perulangan, bahkan lompatan sehingga perlu dilakukan perubahan urutan eksekusi yang disebut sebagai “kendali program”. Jadi, *programmer* dapat mengatur alur eksekusi program sesuai dengan kebutuhan.

Pada Modul 2 ini kita akan berlatih menggunakan kendali program yang berupa seleksi untuk membuat percabangan. Beberapa pernyataan dalam Bahasa C++ memperbolehkan kita untuk menentukan/menyeleksi berdasarkan suatu kondisi/syarat. Jika kondisi/syarat terpenuhi (bernilai TRUE atau T atau 1), maka sebuah perintah akan dieksekusi. Sebaliknya, jika kondisi tidak terpenuhi (bernilai FALSE atau F atau 0), maka perintah tidak akan dieksekusi. Gambar berikut ini merupakan flowchart dari sebuah pernyataan seleksi.



Pernyataan Kontrol seleksi

Pernyataan IF

Pernyataan if digunakan untuk pengambilan keputusan berdasarkan suatu kondisi tertentu.

Syntax:

```
if(kondisi)
    pernyataan;
```

Pernyataan IF ELSE

Apabila suatu kondisi yang ditentukan benar maka pernyataan dalam kondisi tersebut akan diproses, jika kondisi yang ditentukan salah maka pernyataan lain akan diproses oleh program.

Deklarasi:

```
if(kondisi)
    pernyataan1;
else
    pernyataan2;
```

Pernyataan bersarang if..... else if..... else if..... else if.....

Pernyataan if bersarang digunakan untuk menguji lebih dari satu kondisi. Jika suatu kondisi salah maka kondisi lain akan diuji, jika kondisi yang diuji tersebut salah maka kondisi yang lain lagi akan diuji dan seterusnya.

Deklarasi:

```
if(kondisi1)
    pernyataan1;
else if(kondisi2)
    pernyataan2;
else if(kondisi3)
    pernyataan3;
else if(kondisi4)
    pernyataan4;
else
    pernyataanN;
```

HINT
Pernyataan if-else ini adalah salah satu fungsi kontrol yang penting. Hal ini dibuktikan dengan digunakan hampir pada semua bahasa pemrograman.

Pernyataan switch dan case

Pernyataan switch adalah pernyataan yang digunakan untuk menjalankan salah satu pernyataan dari beberapa kemungkinan pernyataan berdasarkan nilai dari sebuah ungkapan dan nilai penyeleksi.

Deklarasi:

```
switch(ungkapan)
{
case ungkapan1:
    pernyataan1;
    break;
case ungkapan2:
    pernyataan2;
    break;
case ungkapan3:
```

```
        pernyataan3;  
        break;  
        .....  
default:  
        pernyataanN;  
}
```

LAB 2.3: LATIHAN

Berikut adalah kode yang merupakan implementasi dari program penukar angka (swap). Salinlah program tersebut menggunakan text editor, kemudian *compile* dan *run* (eksekusi) program tersebut.

PERCOBAAN 1 : SWAP TWO VALUES

```
#include <iostream>  
  
using namespace std;  
  
int main()  
{  
    int value1, value2, buff;  
    cout << "Masukkan angka 1 : ";  
    cin >> Value1;  
    cout << "Masukkan angka 2 : ";  
    cin >> value2;  
    if(value1 > value 2) {  
        buff = Value1; // memindahkan value 1 ke variabel buff  
        value2 = value1; // memindahkan value 2 ke value 1  
        value2 = buff; // memindahkan isi value 1 yang telah dipindah ke buff ke value 2  
  
        cout << "angka 1 : " << value << endl;  
        cout << "angka 2 : " << value2 << endl;  
        return 0;  
    }  
}
```

1. Apakah terdapat kesalahan pada program tersebut? Bila terdapat kesalahan, cobalah perbaiki program tersebut hingga dapat di-*compile* dan memberikan hasil sesuai yang diharapkan.
2. Apakah hasil/luaran program sudah benar? Jika belum benar, perbaiki.

PERCOBAAN 2 : SWAP WITH POINTER

```
#include <iostream>  
  
int main()  
{  
    int x, y, temp;  
    int *a, *b;  
  
    cout << "Angka 1 : ";
```

```

cin >> x;
cout << "Angka 2 : ";
cin >> y;

if (x<y) {
a = &x;
b = &x;

temp = *a;
*a = *b;
*b = temp;
else
cout<<"Tidak terjadi pertukaran angka";

cout << "Angka 1 setelah swap : " << x;
cout << "\nAngka 2 setelah swap : " << y;

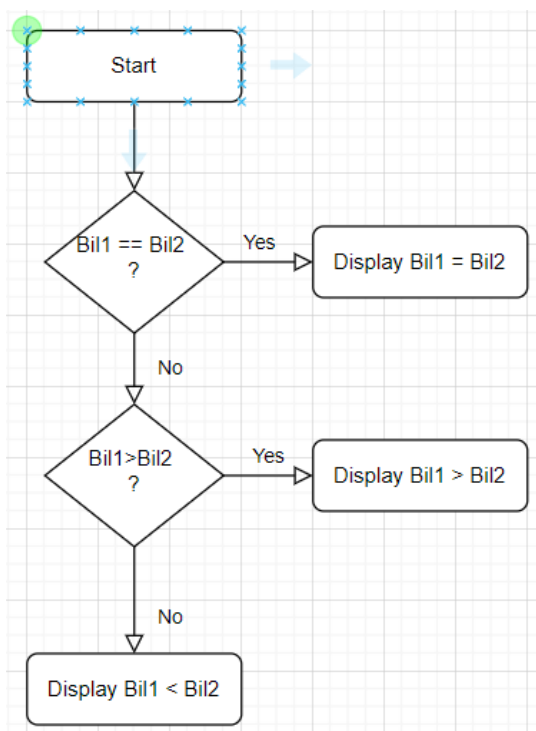
return 0;
}

```

1. Apakah terdapat kesalahan pada program tersebut? Bila terdapat kesalahan, cobalah perbaiki program tersebut hingga dapat di-*compile* dan memberikan hasil.
2. Apakah hasil/luaran program sudah benar? Jika belum benar, perbaiki.

PERCOBAAN 3 : NESTED IF-ELSE

Berikut terlampir *flowchart* sebuah program yang dapat digunakan untuk menentukan relasi antara dua buah bilangan. Relasi yang ada adalah lebih besar, kurang dari, atau sama dengan (>, <, =). Buatlah sebuah kode program dalam C++ berdasarkan *flowchart* di bawah ini!



PERCOBAAN 4 : NESTED IF-ELSE

Berdasarkan tampilan output pada gambar di bawah ini, buatlah program dalam Bahasa C++ dengan menggunakan nested if-else (linear nested selection).

Sample Run 1

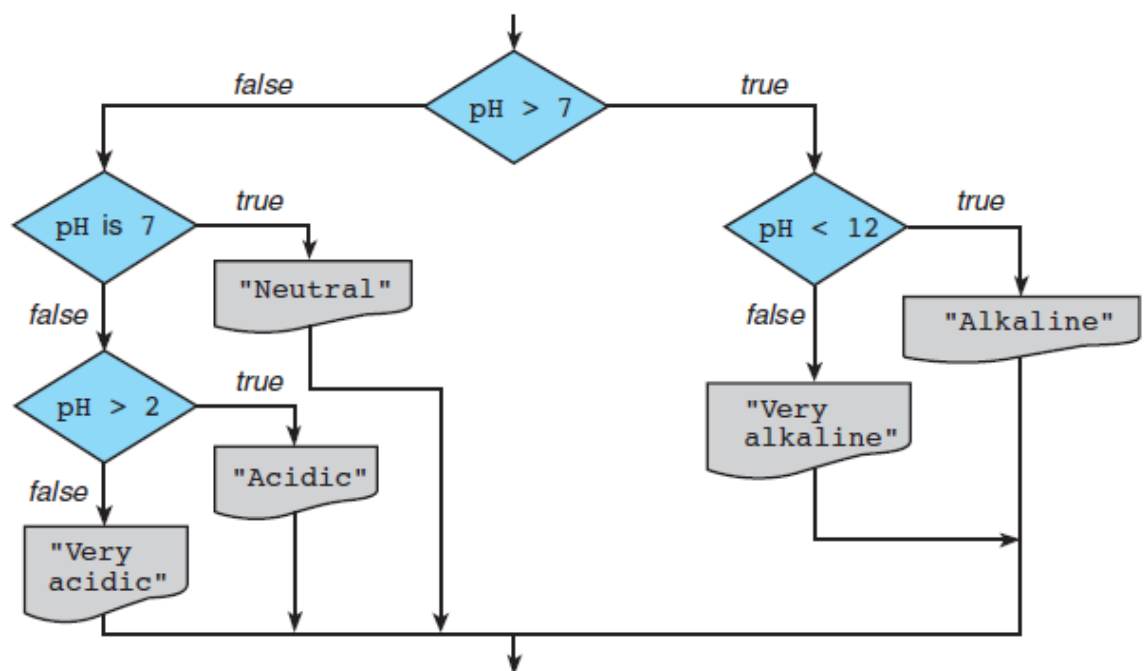
```
Take your resting pulse for 10 seconds.
Enter your pulse rate and press return> 12
Your resting heart rate is 72.
Keep up your exercise program!
```

Sample Run 2

```
Take your resting pulse for 10 seconds.
Enter your pulse rate and press return> 9
Your resting heart rate is 54.
Your heart is in excellent health!
```

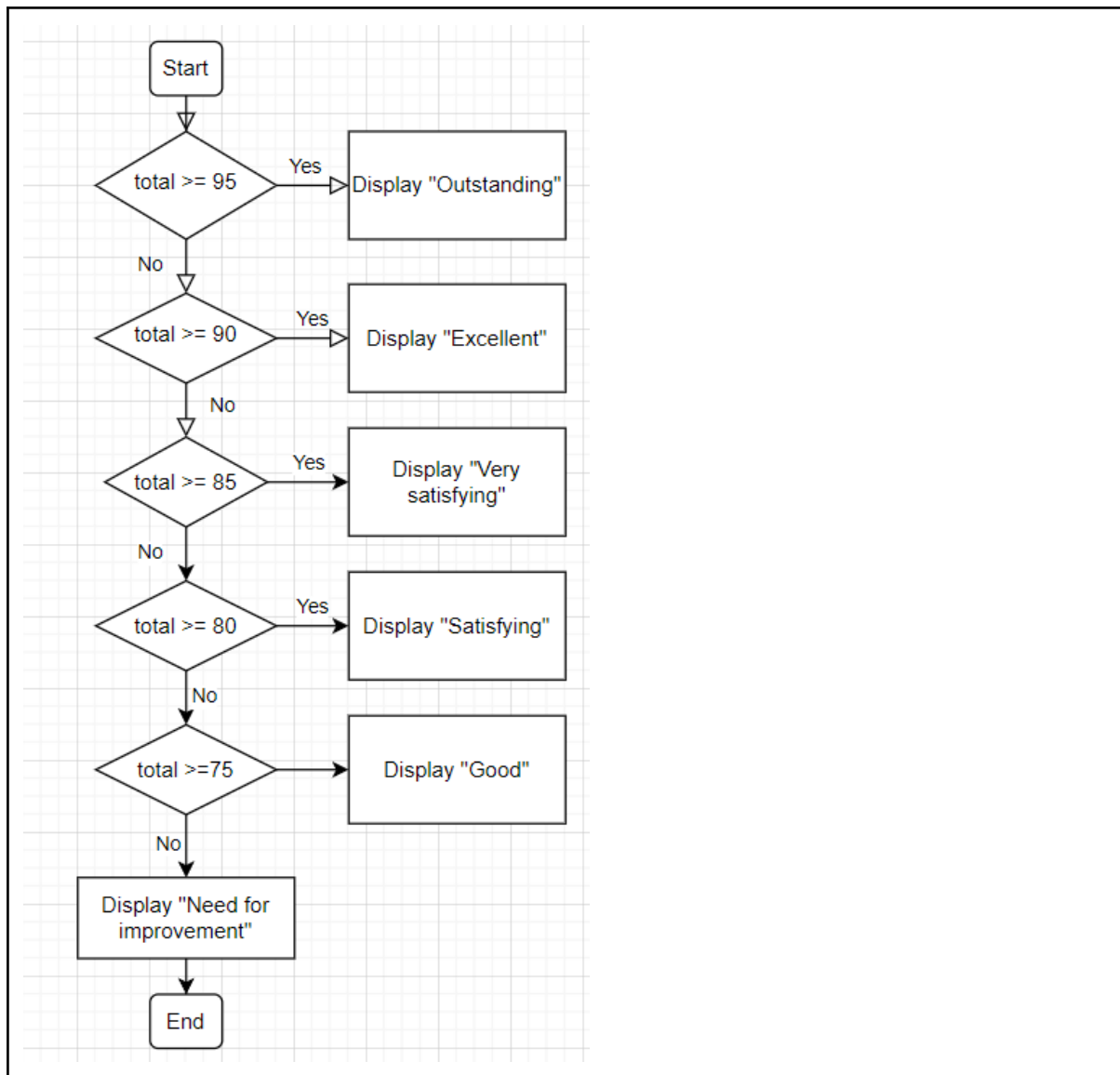
PERCOBAAN 5 : NESTED IF-ELSE

Pelajari algoritma yang direpresentasikan dengan menggunakan flowchart berikut ini. Kemudian implementasikan dengan menggunakan Bahasa C++.



LAB 2.4: CEK POIN 1

Buatlah program dengan menggunakan nested-if untuk multiple selection dalam Bahasa C++ berdasarkan flowchart di bawah ini.



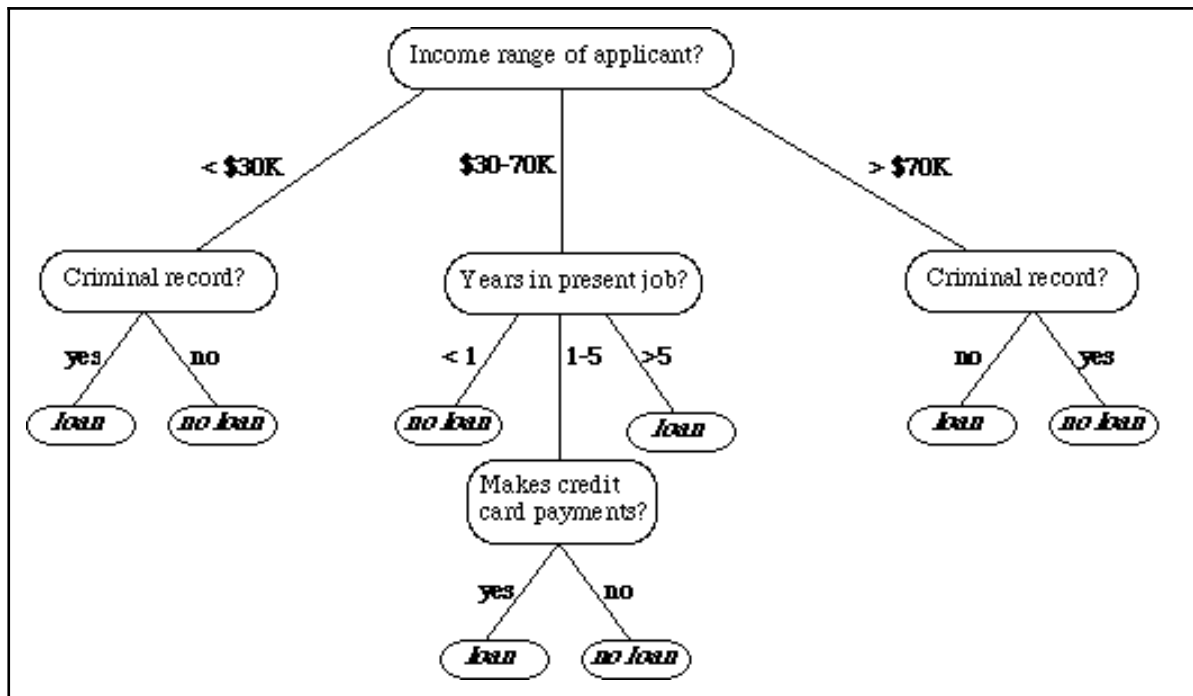
LAB 2.5: CEK POIN 2

Buatlah program dalam Bahasa C++ (dengan menggunakan switch-statement) berdasarkan permasalahan di bawah ini.

Buatlah program agar pengguna dapat memilih untuk menghitung volume dari satu bentuk bangun ruang. Program anda harus menyediakan minimal 3 pilihan bangun ruang.

LAB 2.6: CEK POIN 3

Buatlah program yang menerapkan pernyataan seleksi dalam Bahasa C++ (IF, IF-ELSE, atau SWITCH-CASE) berdasarkan gambar *decision tree* berikut ini.



LAB 2.7: REFERENSI

Di bawah ini adalah referensi online dari UNIT II. Apabila mengalami kesulitan dalam memahami materi, silahkan buka referensi di bawah ini :

1. <http://www.unf.edu/~broggio/cop2221/2221pseu.html>
2. <http://www.jchq.net/certkey/0201certkey.html>
3. <https://www.csharp-console-examples.com/programming-languages/cpp/c-switch-case-statement/>
4. <https://www.programiz.com/cpp-programming/switch-case>
5. <https://www.programiz.com/cpp-programming/if-else>

LAB 2.8: SEKILAS MATERI UNIT BERIKUTNYA

Materi praktikum pemrograman dasar unit selanjutnya akan lebih banyak membahas beberapa hal, terkait dengan algoritma repetisi. Beberapa hal yang patut Anda ketahui, antara lain

1. Pengertian dasar algoritma repetisi.
2. Operasi-operasi dengan algoritma repetisi (WHILE, DO-WHILE, FOR)