

Nama : Ifham Syafwan Fikri
Github : <https://github.com/FikriSyafwan/Penugasan-Praktikum-Pemrograman>
Nomor Mahasiswa : 24/545184/PA/23161
Kelas : KOM B
Dosen Pengampu : Muhammad Husni Santriaji

LAPORAN PRAKTIKUM PEMROGRAMAN

PERTEMUAN 3

Tugas dan Latihan

Number 1: Menghitung determinan dari suatu persamaan

Berikut merupakan hasil akhir dari sistem, apabila bilangan yang diinput adalah 1, 0, dan 1.

Penjelasan selengkapnya sebagai berikut:

Langkah pertama:

- Menuliskan `#include <iostream>` dan `#include <cmath>` karena sistem melibatkan operasi matematika
- Menambah variabel yang akan diberi nilai, yakni Diskriminan, akar1 dan akar2, baganReal, dan imajiner
- Memberi nilai kepada variabel-variabel yang telah ditambahkan
- Menuliskan persamaan yang akan digunakan untuk menyelesaikan permasalahan.

```
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  int main() {
6      double Diskriminan, akar1, akar2, baganReal, imajiner;
7      double a, b, c;
8
9      //memisalkan input bilangan adalah 1, 2, dan 3
10
11     a = 1;
12     b = 0;
13     c = 1;
14
15     //Rumus Diskriminan
16     Diskriminan = b * b - 4 * a * c;
```

Langkah kedua:

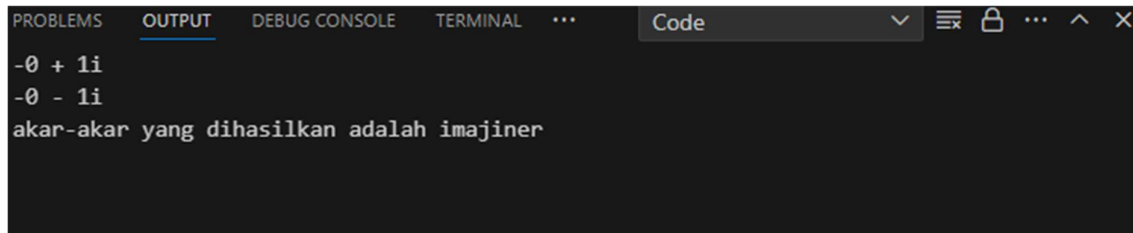
- Membuat pilihan atau *selection* terhadap permasalahan. Determinan > 0 akan memiliki akar-akar real dan berbeda, Determinan $= 0$ hanya akan memiliki satu akar real, dan Determinan < 0 akan menghasilkan akar imajiner.

```
18     if (Diskriminan > 0)
19     {
20
21         //apabila diskriminan lebih dari nol
22         //maka akar-akarnya berupa bilangan real dan
23         //berbeda
24
25         akar1 = (-b + sqrt(Diskriminan)) / (2 * a);
26         akar2 = (-b - sqrt(Diskriminan)) / (2 * a);
27         cout << "akar 1 " << akar1 << endl;
28         cout << "akar 2 " << akar2 << endl;
29         cout << "akar-akar yang dihasilkan adalah real dan berbeda" << endl;
30     }
```

```
32     else if (Diskriminan == 0)
33     {
34
35         //apabila diskriminan sama dengan nol
36         //akarnya hanya satu dan berupa bilangan real
37
38         akar1 = (-b + sqrt(Diskriminan)) / (2 * a);
39         akar2 = (-b - sqrt(Diskriminan)) / (2 * a);
40         cout << "akar 1 " << akar1 << endl;
41         cout << "akar 2 " << akar2 << endl;
42         cout << "akar-akar yang dihasilkan adalah real dan sama" << endl;
43     }
44
45     else {
46
47         //apabila tidak ada opsi yang memenuhi
48         //menunjukkan diskriminan lebih dari nol
49         //maka akar-akarnya berupa bilangan imajiner
50
51         bagianReal = -b / (2 * a);
52         imajiner = sqrt(-Diskriminan) / (2 * a);
53         cout << bagianReal << " + " << imajiner << "i" << endl;
54         cout << bagianReal << " - " << imajiner << "i" << endl;
55         cout << "akar-akar yang dihasilkan adalah imajiner" << endl;
56     }
57
58     return 0;
```

Langkah ketiga:

- Kemudian, setelah menuliskan semuanya, eksekusi hasil dari program.



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL ... Code
-0 + 1i
-0 - 1i
akar-akar yang dihasilkan adalah imajiner
```

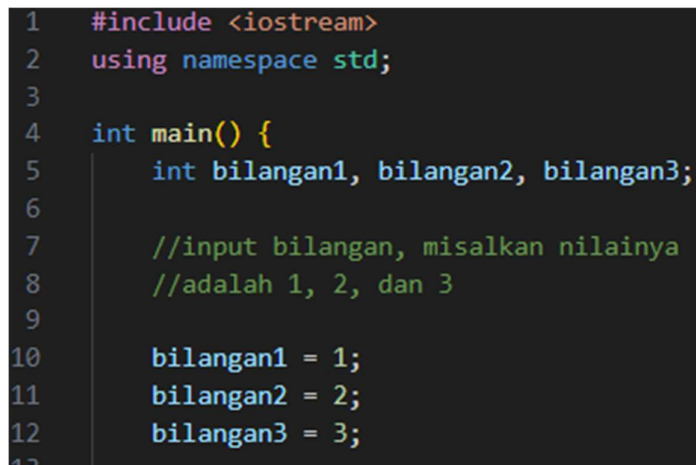
Number 2: Menentukan bilangan yang terbesar

Berikut merupakan hasil akhir sistem, apabila bilangan yang diinput adalah 1, 2, dan 3 di mana bilangan 1 adalah 1, bilangan 2 adalah 2, dan bilangan 3 adalah 3.

Penjelasan selengkapnya sebagai berikut:

Langkah pertama:

- Menuliskan `#include <iostream>` supaya sistem dapat bekerja.
- Menuliskan variabel-variabel untuk menyelesaikan program.
- Memberi nilai kepada variabel-variabel yang telah dibuat



```
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int bilangan1, bilangan2, bilangan3;
6
7      //input bilangan, misalkan nilainya
8      //adalah 1, 2, dan 3
9
10     bilangan1 = 1;
11     bilangan2 = 2;
12     bilangan3 = 3;
13 }
```

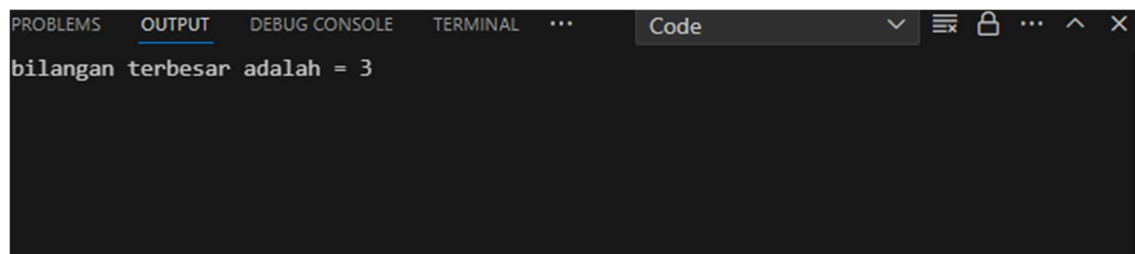
Langkah kedua:

- Membuat pilihan atau selection terhadap permasalahan. Apabila bilangan 1 merupakan lebih dari bilangan 2 dan 3 maka bilangan 1 merupakan bilangan terbesar, apabila bilangan 2 lebih dari bilangan 1 dan 3 maka bilangan 2 merupakan bilangan terbesar, dan begitu pula untuk bilangan 3.

```
14     if (bilangan1 > bilangan2 && bilangan1 > bilangan3) {
15
16         //apabila bilangan 1 lebih besar dari bilangan 2
17         //dan lebih besar dari bilangan 3
18         //maka bilangan 1 merupakan bilangan terbesar
19
20         cout << "bilangan terbesar adalah = " << bilangan1 << endl;
21     }
22
23     else if (bilangan2 > bilangan1 && bilangan2 > bilangan3) {
24
25         //bilangan 2 akan menjadi bilangan terbesar apabila
26         //lebih besar dari bilangan 1 dan bilangan 3
27
28         cout << "bilangan terbesar adalah = " << bilangan2 << endl;
29     }
30
31     else {
32         //Namun, apabila bilangan 1 dan bilangan 2 tidak
33         //memenuhi maka bilangan terbesar adalah
34         //bilangan 3
35
36         cout << "bilangan terbesar adalah = " << bilangan3 << endl;
37     }
38
39 }
```

Langkah ketiga:

- Eksekusi hasil dari program



The screenshot shows a code editor window with a dark theme. The 'OUTPUT' tab is selected, displaying the text 'bilangan terbesar adalah = 3'. The editor interface includes tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', and 'TERMINAL'. The 'Code' tab is also visible, showing the source code. The output text is displayed in a monospaced font.

```
bilangan terbesar adalah = 3
```

Pertanyaan

Question 1: Apa saja instruksi percabangan dalam bahasa pemrograman C++

Pada bahasa pemrograman C++, terdapat enam bentuk percabangan. Berikut merupakan beberapa percabangan yang ada:

1. Percabangan *if*
2. Percabangan *if/else*
3. Percabangan *if/else if/ else*
4. Percabangan *switch/case*
5. Percabangan menggunakan *operator ternary*
6. Percabangan bersarang

Question 2: Jelaskan karakteristik masing-masing jenis perintah tersebut

1. **Percabangan *if*:** Proses percabangan *if* adalah percabangan yang hanya mempunyai satu blok pilihan saat kondisi benar. Percabangan bisa melibatkan satu kondisi, dua kondisi dan tiga kondisi bahkan bisa lebih.
2. **Percabangan *if/else*:** Perbedaan dengan percabangan C++ yang hanya *if* saja adalah di bagian blok kode. Kalau di percabangan *if* hanya bisa satu blok kode, sedangkan *if/else* bisa diisi dengan dua blok kode. Blok pertama untuk kondisi benar dan blok yang kedua untuk yang salah.
3. **Percabangan *if/else if/else*:** Percabangan yang memiliki lebih dari dua blok pilihan. Apabila menggunakan percabangan ini, program akan menjalankan seleksi *if* atau *else if* dengan syarat tertentu yang sudah dimasukkan serta *else* untuk kondisi yang tidak memenuhi persyaratan *if* dan *else if*.
4. **Percabangan *switch/case*:** Percabangan bentuk lain dari *if/else*. Percabangan *switch/case* memudahkan program untuk mengevaluasi satu nilai dengan beberapa kemungkinan hasil tanpa menggunakan banyak perintah *if/else*.
 - a. **Switch** memeriksa nilai variabel atau ekspresi.
 - b. **Case** digunakan untuk menentukan tindakan apa yang akan diambil jika nilai variabel atau ekspresi cocok dengan nilai yang ditentukan dalam case.
 - c. Jika nilai cocok, maka kode di dalam case yang bersangkutan akan dijalankan.
 - d. Jika tidak ada case yang cocok, maka kode dalam **default** (jika ada) akan dijalankan sebagai opsi terakhir.
5. **Percabangan menggunakan operator ternary:** Percabangan ini merupakan bentuk singkat dari pernyataan *if/else* yang hanya digunakan ketika terdapat dua kemungkinan hasil, yaitu kondisi benar dan salah.

Kondisi ? hasil_jika_benar : hasil_jika_salah

- a. **kondisi:** Syarat atau ekspresi yang akan diuji (bernilai *true* atau *false*).
 - b. **hasil_jika_benar:** Nilai atau pernyataan yang dijalankan jika kondisi bernilai *true*.
 - c. **hasil_jika_salah:** Nilai atau pernyataan yang dijalankan jika kondisi bernilai *false*.
6. **Percabangan bersarang:** Percabangan bersarang adalah struktur percabangan yang menempatkan satu atau lebih pernyataan *if*, *switch*, atau *if/else* di dalam blok percabangan lainnya. Dengan kata lain, percabangan dalam percabangan.