

MI2101

Praktikum Teknik Komputasi

Modul 1

Fadjar Fathurrahman

2018

1 Tujuan

- Dapat membuat project C/C++ sederhana dengan menggunakan CodeBlocks
- Dapat membuat kode sumber C/C++ dengan menggunakan CodeBlocks atau editor teks
- Dapat melakukan proses build project C/C++ pada CodeBlocks
- Mengenal baris perintah (*command line* pada Linux) serta beberapa perintah sederhana
- Dapat melakukan kompilasi program dengan menggunakan baris perintah pada Linux
- Mengenal beberapa kesalahan umum pada pembuatan program serta solusinya
- Dapat membuat program C++ sederhana untuk melakukan perhitungan matematika sederhana

2 Perangkat lunak yang diperlukan

- Linux OS
- CodeBlocks yang telah dikonfigurasi untuk kompiler GNU C/C++
- Terminal emulator dengan *bash* sebagai shell (baris perintah)
- Editor teks seperti gedit

3 Petunjuk dasar

Buat direktori (folder) dengan format Modul01_NIM_Nama. Semua pekerjaan Anda akan dilakukan dalam direktori ini.

Jangan gunakan spasi jika nama Anda lebih dari satu kata

Gunakan underscore `_` sebagai pengganti spasi

Contoh:

```
Modul01_Jaka_Sembung (benar)
Modul01_Jaka Sembung (salah)
Modul01 Jaka Sembung (salah)
```

4 Program "Hello World" dengan menggunakan CodeBlocks

Buatlan sebuah proyek kosong pada CodeBlocks dengan nama Hello01 dengan langkah sebagai berikut:

1. Pada menu New → Project → Empty Project
2. Konfigurasi project:
 - Project Title: Hello01
 - Folder to create project in: /home/students/Modul01_Jaka_Sembung/
 - Project filename: Hello01.cbp
 - Resulting filename:
/home/students/Modul01_Jaka_Sembung/Hello01/Hello01.cbp
 - Kemudian klik Next

Next.

3. Pastikan bahwa yang digunakan adalah Compiler GCC ceklis semua, jangan ubah konfigurasi. Kemudian Finish
4. Tambahkan file baru: menu New → Files → Category C/C++ source → C++ → Masukkan path file → Add file to active projects. Ceklis semua pilihan Debug/Release dan Finish. Nama file: Hello01.cpp dengan isi sebagai berikut

```
// Nama:
// NIM:

#include <iostream>
using namespace std;

int main()
{
    cout << "Hello World" << endl;
    return 0;
}
```

Kemudian Build (F9). Pastikan bahwa tidak ada kesalahan pada saat proses ini.

4.1 Tugas

1. Tuliskan output dari program ketika dijalankan.
2. Tuliskan output dari Build Log dan Build messages.
3. Berikut ini adalah struktur dari direktori Hello01 ketika Anda telah berhasil membangun dan menjalankan proyek Hello01 ¹:

```
Hello01
├── bin
│   └── Debug
│       └── Hello01
├── Hello01.cbp
├── Hello01.cpp
├── Hello01.depend
└── Hello01.layout
```

¹Pada beberapa kasus, file *.depend dan *.layout tidak ada pada direktori Anda. Tidak perlu khawatir karena hal tersebut tidak penting dalam praktikum ini.

```

└─ obj
    └─ Debug
        └─ Hello01.o

```

Buat file baru dengan nama `Hello02.cpp` dalam direktori ini sehingga struktur direktorinya menjadi sebagai berikut:

```

Hello01/
├─ bin
│   └─ Debug
│       └─ Hello01
├─ Hello01.cbp
├─ Hello01.cpp
├─ Hello01.depend
├─ Hello01.layout
├─ Hello02.cpp    <--- File yang baru dibuat
└─ obj
    └─ Debug
        └─ Hello01.o

```

Isi file `Hello02.cpp` ini sebagai berikut:

```

#include <iostream>
using namespace std;
int main()
{
    cout << "Hello this is MI2101" << endl;
    return 0;
}

```

Kemudian coba build kembali project yang Anda buat. Apa yang terjadi ? Tuliskan keluaran dari program apabila project ini berhasil dibangun. Jika terjadi kesalahan, apa yang harus dilakukan?

4. Tutup semua jendela CodeBlocks. Sekarang buka File Explorer di Ubuntu, kemudian buka direktori di mana project CodeBlocks ini Anda buat. Bandingkan dua kasus berikut.

- CodeBlocks digunakan untuk membuka file `Hello01.cpp`
- CodeBlocks digunakan untuk membuka file `Hello01.cbp`

(Untuk masing-masing kasus CodeBlocks harus ditutup terlebih dahulu) Adakah perbedaan yang Anda amati? Apakah perbedaan antara `Hello01.cpp` dengan `Hello01.cbp` ?

5 Program 'Hello World' dengan menggunakan baris perintah

Buka Terminal (program khusus untuk menjalankan baris perintah).

Perhatikan demonstrasi yang dilakukan di depan kelas.

Catat perintah-perintah yang telah didemonstrasikan dalam tabel berikut ini.

Perintah	Penjelasan mengenai perintah (keluaran dan/atau apa yang dilakukan)
ls pwd dan seterusnya	

Lengkapi langkah-langkah (dan perintah yang digunakan pada terminal) berikut ini yang diperlukan untuk menjalankan program `Hello01`. Kita akan menggunakan file `Hello01.cpp` yang sudah kita buat pada CodeBlocks sebelumnya.

1. Cek direktori kerja awal ketika membuka terminal. Perintah yang diperlukan adalah Direktori kerja awal adalah
2. Tentukan lokasi file .cpp yang ingin dibangun (build = compile + link). Lokasi file tersebut adalah
3. Ubah direktori kerja ke direktori yang berisi file .cpp yang telah disebutkan sebelumnya. Direktori tersebut adalah Perintah yang diperlukan adalah
4. Pastikan bahwa perintah yang dilakukan pada langkah 3 sukses dan kita tidak melakukan kesalahan dengan cara mengecek direktori kerja sekarang. Perintah yang diperlukan adalah Direktori kerja sekarang adalah
5. Pastikan bahwa file .cpp tersebut ada pada direktori kerja. Hal ini dapat dilakukan dengan cara melihat daftar file dan subdirektori apa saja yang ada pada direktori saat ini. Perintah yang diperlukan adalah File-file yang ada pada direktori kerja adalah
6. Bangun (Build) program dengan menggunakan perintah
7. Pastikan proses pada langkah sebelumnya telah berhasil (tidak ada kesalahan atau pesan error). Cek apakah ada file baru yang dihasilkan. Jika ada file tersebut bernama
8. Eksekusi program yang dihasilkan pada terminal. Perintah yang diperlukan adalah Hasil/output dari eksekusi program

6 Perhitungan arimatika sederhana

Buat project CodeBlocks baru dengan nama Aritmatika01 dan tambahkan file C++ pada proyek tersebut dengan nama aritmatika01.cpp.

Alternatif Untuk Anda yang lebih senang bekerja dengan terminal, buat direktori Aritmatika01 dan file aritmatika01.cpp.

Berikut ini adalah kode sumber untuk aritmatika01.cpp.

```
// Nama:
// NIM:
#include <iostream>
using namespace std;

int main()
{
    int a, b, c;

    a = 123;
    b = 456;

    // Tampilkan nilai a and b ke layar
    cout << "a = " << a << endl;
    cout << "b = " << b << endl;

    // Lakukan perhitungan tambah, kurang, kali, dan bagi
    c = a + b;
    cout << "a + b = " << c << endl;

    c = a - b;
    cout << "a - b = " << c << endl;

    c = a * b;
```

```

    cout << "a * b = " << c << endl;

    c = a / b;
    cout << "a / b = " << c << endl;

    return 0;
}

```

6.1 Tugas

1. Tuliskan keluaran dari program tersebut ketika dijalankan.
2. Apakah ada hasil perhitungan yang Anda anggap salah ? Bagaimana cara memperbaikinya ?

7 Latihan dan Eksplorasi (Bagian A)

Pada kerjakan latihan ini dengan menggunakan CodeBlocks atau baris perintah pada terminal.

Untuk setiap percobaan, serta berikan penjelasan dan/atau komentar Anda.

Cek apakah kode berikut ini dapat dibuild atau dijalankan tanpa ada kesalahan (*syntax error*).

Tuliskan pesan kesalahan (jika ada) yang tampil pada Build Log (jika menggunakan CodeBlocks) atau terminal, jelaskan kesalahan yang terjadi kemudian perbaiki.

1. Kode sumber

```

#include <iostream>
using namespace std;
int main()
{
    cout << "Hello World" << endl
    return 0;
}

```

2. Kode sumber

```

#include <iostream>
using namespace std;
int main()
{
    cout < "Hello World" < endl
    return 0;
}

```

3. Kode sumber

```

using namespace std;
int main()
{
    cout << "Hello World" << endl;
    return 0;
}

```

4. Kode sumber:

```
#include <iostream>
int main()
{
    cout << "Hello World" << endl;
    return 0;
}
```

5. Kode sumber:

```
#include <iostream>
int main()
{
    std::cout << "Hello World" << std::endl;
    return 0;
}
```

6. Kode sumber:

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Hello World" << endl;
}
```

Referensi berikut ini boleh dibaca:

- <https://www.codeproject.com/Questions/693038/why-do-we-have-to-use-return>
- <http://www.cplusplus.com/forum/beginner/24461/>

7. Kode sumber

```
#include <stdio.h>
int main()
{
    printf("Hello World\n");
    return 0;
}
```

8. Kode sumber

```
#include <stdio.h>
int main()
{
    printf("Hello World\n");
    return 0;
}
```

8 Latihan dan eksplorasi (Bagian B)

1. Modifikasi program pada proyek Aritmatika01 agar menghasilkan output sebagai berikut (dengan menggunakan nilai a dan b yang sama)

```
123 + 456 = 579
123 - 456 = -333
123 * 456 = 56088
123 / 456 = 0.269737
```

2. Lengkapi program berikut ini. Program ini bertujuan untuk mencari akar dari persamaan kuadrat $ax^2 + bx + c = 0$, yaitu:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (1)$$

Untuk sementara, program ini hanya memperhitungkan kasus nilai diskriminan $D = b^2 - 4ac \geq 0$. Kasus yang lebih umum akan dipelajari di Modul 2.

```
#include <iostream>
#include <cmath> // diperlukan untuk sqrt
using namespace std;
int main()
{
    ....

    a = 1.2;
    b = 1.3;
    c = -4.0

    // Hitung diskriminan
    .... d;
    d = ....;
    // Tampilkan nilai diskriminan
    cout << ....

    float x1, x2;
    x1 = ....;
    x2 = ....;

    // Tampilkan nilai x1 dan x2
    cout << "x1 = " << x1 << endl;
    cout << "x2 = " << x2 << endl;

    return 0;
}
```