

Lembar Kerja
Praktikum Pemrograman Dasar



Nama Praktikan	Adi Febriana	NPM 140910200022
Tanggal Praktikum	15 Oktober 2020	
Anggota Kelompok	Lidia Marsarlina	NPM 140910200021
	Adi Febriana	NPM 140910200022
	Fata Hibrizi Thufail Wijdan	NPM 140910200023
	Frizqi Ramadhandika Listanto	NPM 140910200024
	Athaya Salsabil	NPM 140910200025
Asisten	Rafly	
Modul	Modul 1 Pendahuluan (1.3.8 - 1.3.13)	

PROGRAM STUDI TEKNIK ELEKTRO
UNIVERSITAS PADJADJARAN

2020

LEMBAR PENGESAHAN

Nama Praktikan	Adi Febriana	NPM 140910200022
Tanggal Praktikum	15 Oktober 2020	
Asisten	Rafly	
Modul	Modul 1 Pendahuluan (1.3.8 - 1.3.13)	
Tujuan	Setelah mengikuti praktikum dengan pokok bahasan Pengenalan Bahasa C, mahasiswa diharapkan mengetahui dan memahami konsep-konsep dasar pemrograman C dengan cara mampu membuat program C yang sederhana, mampu menggunakan statement input dan output yang sederhana, mengenal tipe-tipe data dalam C, dan mampu memilih serta menggunakan berbagai macam operator	

Nilai	Asisten

Modul 1

Pokok Bahasan

1. Katakunci dan pengenalan (Keywords and Identifier)
2. Variabel dan konstanta (Variables and Constants)
3. Tipe data (Data Types)
4. Input/Output
5. Operator pemrograman C (Programming Operators)

Tujuan

Setelah mengikuti praktikum dengan pokok bahasan Pengenalan Bahasa C, mahasiswa diharapkan mengetahui dan memahami konsep-konsep dasar pemrograman C dengan cara mampu membuat program C yang sederhana, mampu menggunakan statement input dan output yang sederhana, mengenal tipe-tipe data dalam C, dan mampu memilih serta menggunakan berbagai macam operator

Percobaan

1.1 Listing Program / Script program

```
1 // Adi Febriana (140910200022)
2
3 #include <stdio.h>
4 #include <conio.h>
5
6 main()
7 {
8     float nilai1, nilai2, hasil_kali;
9     printf("Masukkan dua bilangan riil: ");
10    scanf("%f %f", &nilai1, &nilai2);
11    hasil_kali = (10+nilai1)*(20+nilai2);
12    printf("Hasil operasi dengan format standar = %f\n", hasil_kali);
13    printf("Hasil operasi dengan format 12.2 = %12.2f\n", hasil_kali);
14    getch();
15    return 0;
16 }
```

Message

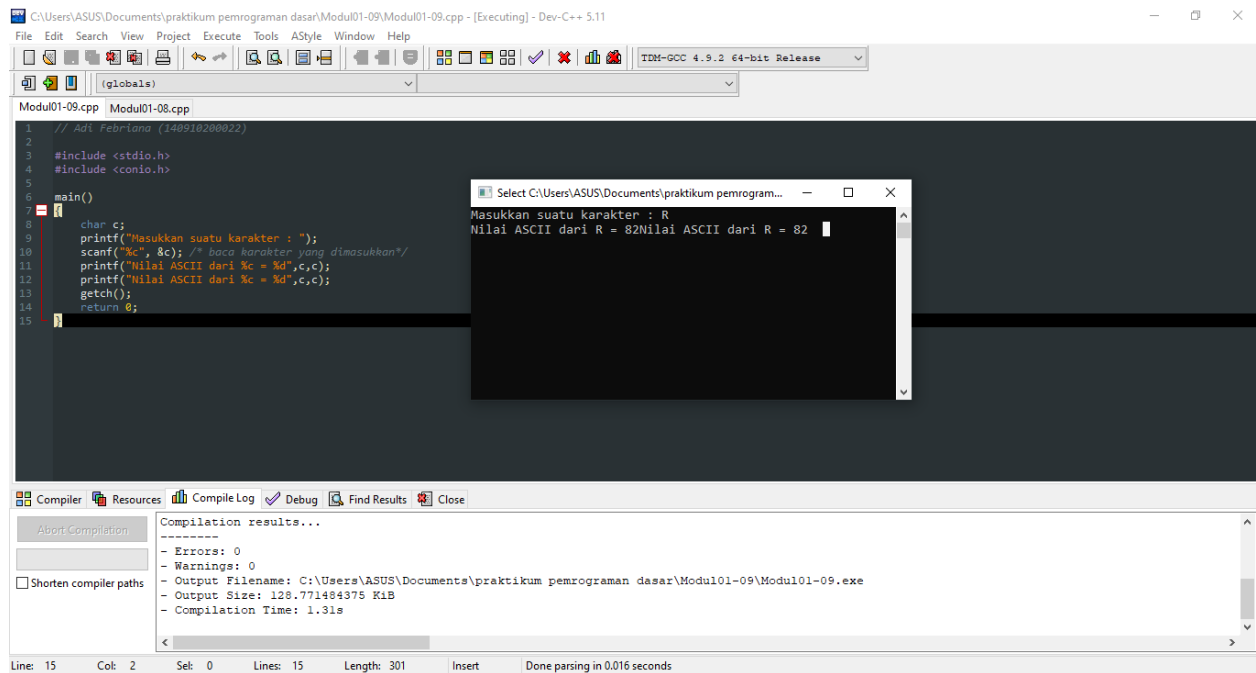
Line	Col	File	Message
		C:\Program Files (x86)\Dev-Cpp\MinGW64\x86_64-w64-...	cannot open output file C:\Users\ASUS\Documents\praktikum pemrograman dasar\modul 1.3.8\Modul01-08.exe: Permission denied
		C:\Users\ASUS\Documents\praktikum pemrograman da...	[Error] ld returned 1 exit status

Line: 5 Col: 1 Sel: 0 Lines: 16 Length: 403 Insert Done parsing in 0.047 seconds

Analisis :

Pada baris delapan, data 2 bilangan riil dari keyboard user di input dan dimasukkan kedalam variabel nilai1 dan nilai 2, nilai yang dimasukkan disini ialah 1 untuk nilai1 dan 2 untuk nilai2. Lalu variabel nilai1 di tambah dengan sepuluh sehingga $10 + 1 = 11$, dan variabel nilai2 ditambah dengan 20 sehingga $20 + 2 = 22$. Lalu nilai1 dikalikan dengan nilai2, $11 \times 22 = 242$. Hasilnya akan tampak di konsol pada baris 3 dan 4. Ada perbedaan cara penyajian outputnya, pada baris 3 dikonsol, hasilnya 242.000000 dengan 6 digit dibelakang koma, formater yang digunakan ialah %f yang berfungsi untuk membaca bilangan real. Alalu pada baris 4 di konsol, hasilnya 242.00, namun terdapat jarak sebesar 12 karakter dari karakter “=”. Jika kita lihat pada baris 11 di Dev Cpp, formater yang digunakan %f juga. Namun, diantara karakter “%” dan “f”, terdapat 12.2 yang artinya, menjorok kekanan sebanyak 12 karakter dan 2 digit angka dibelakang koma.

1.2 Listing Program / Script program



The screenshot shows a C++ IDE with a source code editor, a compiler window, and a console window. The source code is as follows:

```
1 // Adi Febrina (140910200022)
2
3 #include <stdio.h>
4 #include <conio.h>
5
6 main()
7 {
8     char c;
9     printf("Masukkan suatu karakter : ");
10    scanf("%c", &c); /* baca karakter yang dimasukkan */
11    printf("Nilai ASCII dari %c = %d", c, c);
12    printf("Nilai ASCII dari %c = %d", c, c);
13    getch();
14    return 0;
15 }
```

The compiler window shows the following output:

```
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\ASUS\Documents\praktikum pemrograman dasar\Modul01-09\Modul01-09.exe
- Output Size: 128.771484375 KiB
- Compilation Time: 1.31s
```

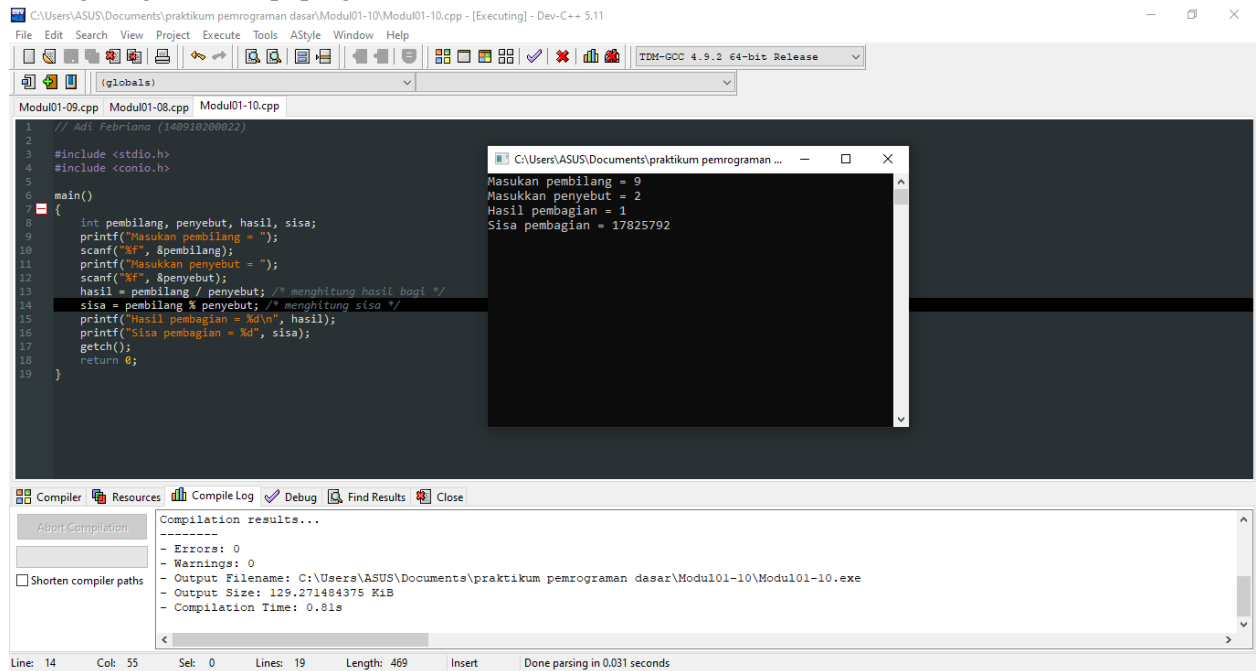
The console window shows the following output:

```
Masukkan suatu karakter : R
Nilai ASCII dari R = 82Nilai ASCII dari R = 82
```

Analisis :

Pada baris 8, dideklarasikan variabel *c* dengan tipe datanya karakter. Lalu pada baris 9, data di input dengan fungsi `scanf()`, dan data pun di *output* pada baris 11 dan 12 dengan fungsi `printf()`. Pada baris 11 dan 12, data karakter yang diinput diubah menjadi desimal dengan menggunakan formater `%d`. Angka desimal yang telah diubah, berasal dari tabel ASCII (American Standard Code for Information Interchange). Angka desimal yang terdapat di tabel ASCII dimulai dari 0 hingga 127 dan untuk karakter dimulai dari [NULL], [BACKSPACE], simbol-simbol seperti “,”, “\$”, dan lainnya, serta hingga huruf kapital dan huruf kecil. Pada gambar diatas, ketika “R” dimasukkan, nilai ASCII 82 akan disimpan dimemori bukan huruf “R”nya.

1.3 Listing Program / Script program



The screenshot shows a C++ IDE with a file named `Modul01-10.cpp`. The code is as follows:

```
1 // Adi Febrina (140910200022)
2
3 #include <stdio.h>
4 #include <conio.h>
5
6 main()
7 {
8     int pembilang, penyebut, hasil, sisa;
9     printf("Masukan pembilang = ");
10    scanf("%i", &pembilang);
11    printf("Masukkan penyebut = ");
12    scanf("%i", &penyebut);
13    hasil = pembilang / penyebut; /* menghitung hasil bagi */
14    sisa = pembilang % penyebut; /* menghitung sisa */
15    printf("hasil pembagian = %d\n", hasil);
16    printf("Sisa pembagian = %d", sisa);
17    getch();
18    return 0;
19 }
```

The output window shows the following results:

```
Masukan pembilang = 9
Masukkan penyebut = 2
Hasil pembagian = 4
Sisa pembagian = 1
```

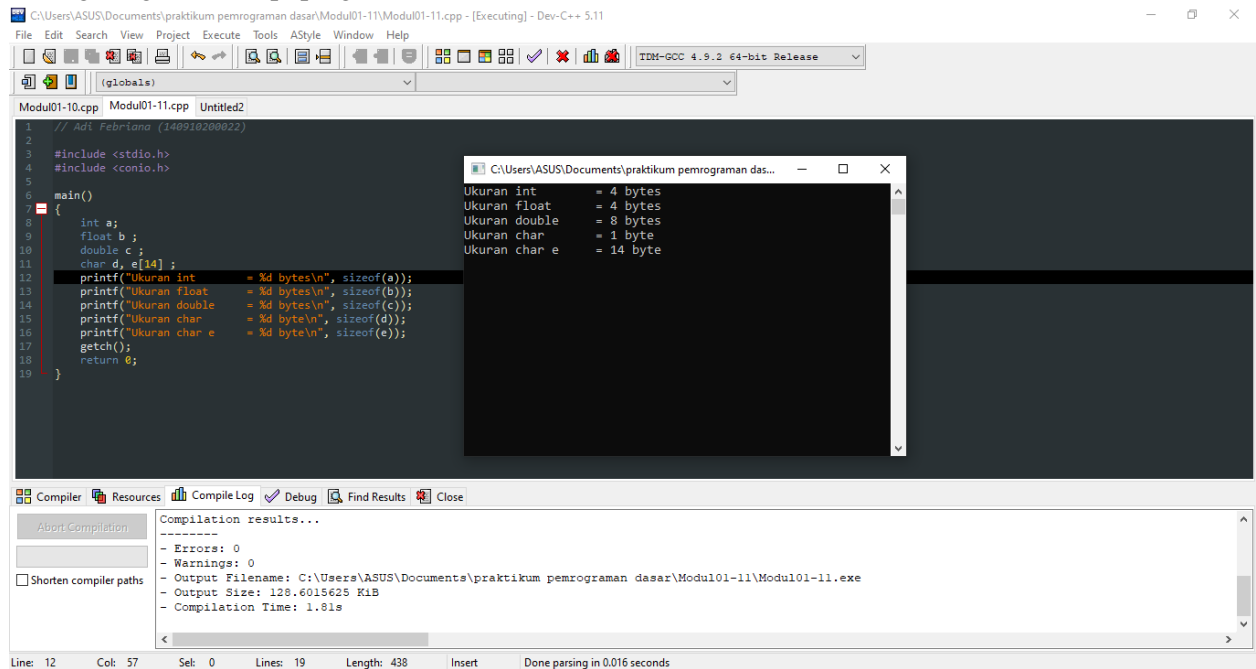
The compiler window shows the following compilation results:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\ASUS\Documents\praktikum pemrograman dasar\Modul01-10\Modul01-10.exe
- Output Size: 129.271404375 KiB
- Compilation Time: 0.81s
```

Analisis :

Pada baris 8, dideklarasikan variabel pembilang, penyebut, hasil, dan sisa, dengan tipe data integer. Pada baris 10 data pembilang diinput dan pada baris 12 diinput data penyebut. Lalu pada baris 13 simbol “/” berfungsi sebagai pembagian dan pada baris 14 simbol “%” berfungsi sebagai sisa bagi. Sisa bagi adalah sisa dari hasil suatu pembagian, seperti gambar diatas, hasil dari 9 dibagi 2 adalah 4 dengan sisa 1. Pada hasil percobaan kami, ketika baris 10 dan 12 diubah menjadi formater %f, hasil bagi dan sisa bagi menjadi salah. Seharusnya hasil sisa akan menampilkan 4 (pada integer, digit dibelakang koma akan diabaikan), dan sisa bagi akan menampilkan 1.

1.4 Listing Program / Script program



The screenshot shows a C++ IDE with a source file named `Modul01-11.cpp` and a terminal window displaying the output of the program. The source code defines variables `a` (int), `b` (float), `c` (double), `d` (char array), and `e` (char), and uses `sizeof()` to determine their memory sizes. The terminal output shows the results: `Ukuran int = 4 bytes`, `Ukuran float = 4 bytes`, `Ukuran double = 8 bytes`, `Ukuran char = 1 byte`, and `Ukuran char e = 14 byte`. The IDE also shows a compilation log with 0 errors and 0 warnings, and a compilation time of 1.81s.

```
// Adi Febriana (140910200022)
#include <stdio.h>
#include <conio.h>

main()
{
    int a;
    float b;
    double c;
    char d, e[14];
    printf("Ukuran int = %d bytes\n", sizeof(a));
    printf("Ukuran float = %d bytes\n", sizeof(b));
    printf("Ukuran double = %d bytes\n", sizeof(c));
    printf("Ukuran char = %d bytes\n", sizeof(d));
    printf("Ukuran char e = %d bytes\n", sizeof(e));
    getch();
    return 0;
}
```

Compilation results...

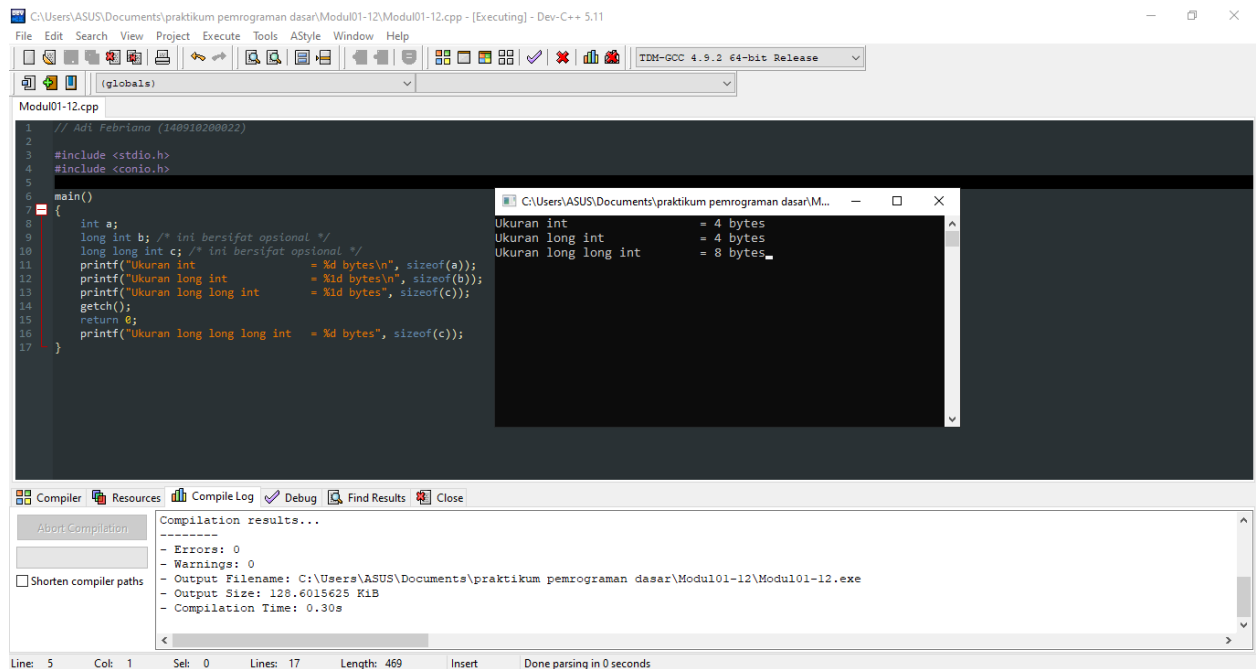
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\ASUS\Documents\praktikum pemrograman dasar\Modul01-11\Modul01-11.exe
- Output Size: 120.6015625 KiB
- Compilation Time: 1.81s

Line: 12 Col: 57 Sel: 0 Lines: 19 Length: 438 Insert Done parsing in 0.016 seconds

Analisis :

Pada baris 8 hingga 11 dideklarasikan variabel `a`, `b`, `c`, `d`, dan `e`, dengan masing-masing tipe datanya integer, float, double dan karakter. Operator `sizeof()` yang digunakan pada baris 12 hingga 16 berfungsi untuk mengungkap berapa banyak memori yang digunakan oleh variabel. Setiap tipe data membutuhkan ruang memori yang berbeda, tipe data integer membutuhkan 4 bytes, float 4 bytes, double 8 bytes, dan karakter 1 byte. Besar ruang memori ini dapat diubah menggunakan simbol “[...]” yang diletakkan disebelah nama variabel seperti pada baris 11. Didalam simbol “[...]” dimasukkan nilai integer, nilai tersebut akan dikalikan dengan besar memori tipe datanya. Misalnya pada baris 11, besar memori variabel `e` yang sebelumnya 1 byte lalu dikalikan dengan 14, besar memorinya menjadi 14 byte.

1.5 Listing Program / Script program



```
1 // Adi Febrina (14091020022)
2
3 #include <stdio.h>
4 #include <conio.h>
5
6 main()
7 {
8     int a;
9     long int b; /* ini bersifat opsional */
10    long long int c; /* ini bersifat opsional */
11    printf("Ukuran int = %d bytes\n", sizeof(a));
12    printf("Ukuran long int = %d bytes\n", sizeof(b));
13    printf("Ukuran long long int = %d bytes", sizeof(c));
14    getch();
15    return 0;
16    printf("Ukuran long long long int = %d bytes", sizeof(c));
17 }
```

Ukuran int = 4 bytes
Ukuran long int = 4 bytes
Ukuran long long int = 8 bytes

Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\ASUS\Documents\praktikum pemrograman dasar\Modul01-12\Modul01-12.exe
- Output Size: 128.6015625 KiB
- Compilation Time: 0.30s

Analisis :

Pada baris 8 hingga baris 10, di deklarasikan variabel a, b, dan c, dengan tipe data integer. Namun, untuk baris 9 ditambahkan “long” di depan “int” dan baris 10 ditambahkan “long long” di depan “int”. long sendiri berfungsi sebagai pengubah ukuran variabel menjadi lebih besar. Tampak di hasil ukuran int itu 4 bytes, long int 4 bytes, dan long long int 8 bytes. Int bisa berukuran 2 bytes atau 4 bytes maka dari itu ukuran int bisa sama dengan long int. Untuk baris 16 tidak akan ditampilkan di hasil output, karena posisinya berada di bawah return 0, yang dimana semua eksekusi kode berakhir di return 0. Jika di pindahkan ke atas getch(), maka akan tampil di outputnya.

1.6 Listing Program / Script program

The screenshot shows a C++ IDE with a source code editor and a console window. The source code is as follows:

```
1 // Adi Febrina (14091020022)
2
3 #include <stdio.h>
4 #include <conio.h>
5
6 main()
7 {
8     float a, b, temp;
9     printf("Masukkan nilai a: ");
10    scanf("%f", &a);
11    printf("Masukkan nilai b: ");
12    scanf("%f", &b);
13    temp = a; /* Nilai a tersimpan di var temp */
14    a = b; /* Nilai b tersimpan di var a */
15    //b = temp; /* Nilai temp*/
16    printf("\nSetelah penukaran, nilai a = %.2f\n", a);
17    printf("Setelah penukaran, nilai b = %.2f", b);
18    getch();
19    return 0;
20 }
```

The console window shows the following output:

```
Masukkan nilai a: 3
Masukkan nilai b: 2
Setelah penukaran, nilai a = 2.00
Setelah penukaran, nilai b = 2.00
```

Below the console window, the compiler output is displayed:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\ASUS\Documents\praktikum pemrograman dasar\Modul01-13\Modul01-13.exe
- Output Size: 129.271484375 KiB
- Compilation Time: 1.89s
```

The status bar at the bottom indicates: Line: 5, Col: 1, Sek: 0, Lines: 20, Length: 462, Insert, Done parsing in 0.016 seconds.

Analisis :

Pada baris 8, dideklarasikan variabel a, b, dan temp, dengan tipe data float. Pada baris 10 dan 12 dilakukan pemasukkan data dari keyboard user ke variabel a dan b. pada baris 13 hingga 15, proses penukaran isi dua variabel. Pada baris 13, nilai variabel a akan disimpan ke variabel sementara atau variabel temp, lalu nilai variabel b akan disimpan ke variabel a, dan yang terakhir, nilai dari variabel temp akan disimpan di variabel b, sehingga isi variabel a dan b akan bertukar. Jika baris 15 dihapus, maka yang terjadi nilai dari variabel a tidak akan tampil di output, karena nilai variabel a yang dimana di pindahkan ke variabel temp, tidak tersimpan di variabel b. sehingga yang tampil di output hanya akan nilai variabel b saja.

Kesimpulan

Jadi, setelah melakukan praktikum modul 1 dari 1.3.8 hingga 1.3.13, praktikan mengetahui dan memahami konsep dasar bahasa pemrograman C dan mampu membuat program C sederhana seputar yang di praktekkan diatas, seperti operasi input/output dan perkalian dua bilangan riil, mencari nilai ASCII yang di input oleh pengguna/*user*, Operasi sisa dan hasil bagi, menggunakan operator sizeof untuk menghitung ukuran variabel, menggunakan operator pengubah ukuran variabel, dan menukar isi dua variabel.