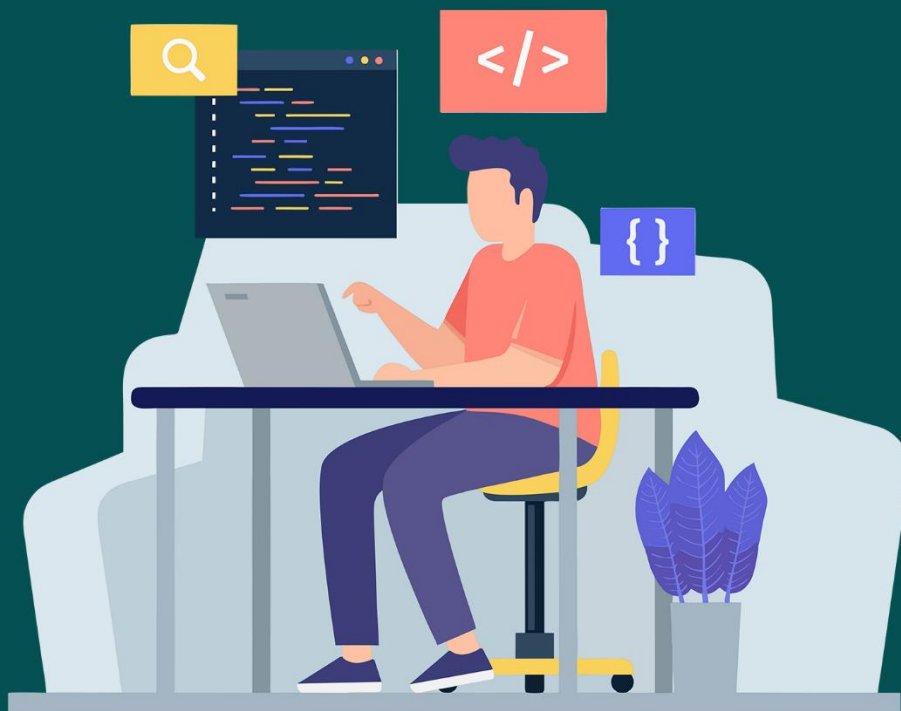


Modul Praktikum
Algoritma dan Pemrograman I

PENGENALAN VARIABEL, OPERASI, & MEKANISME I/O



TIM ASISTEN PEMROGRAMAN
ANGKATAN 12

Departemen Pendidikan Ilmu Komputer
Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam
Universitas Pendidikan Indonesia
2021



PENDAHULUAN

Algoritma adalah solusi dengan runtutan instruksi yang jelas untuk memecahkan suatu masalah. Algoritma yang ditulis dalam bahasa komputer dinamakan **program**. Bahasa komputer yang digunakan dalam menulis program dinamakan **bahasa pemrograman**. Orang yang membuat program komputer disebut **pemrogram**, dan kegiatan merancang dan menulis program disebut **pemrograman**. Di dalam pemrograman ada aktivitas menulis kode program, kegiatan ini dinamakan **coding**.

Algoritma di dalam pemrograman, merupakan solusi dari suatu masalah yang harus dipecahkan dengan menggunakan komputer, harus dibuat secara runtut agar komputer mengerti dan mampu mengeksekusinya.

Topik yang Dipelajari

1. Menulis program dalam bahasa C
2. Kompilasi
3. Variabel dan tipe data
4. Operator Aritmatika
5. Mekanisme input dan output (I/O)

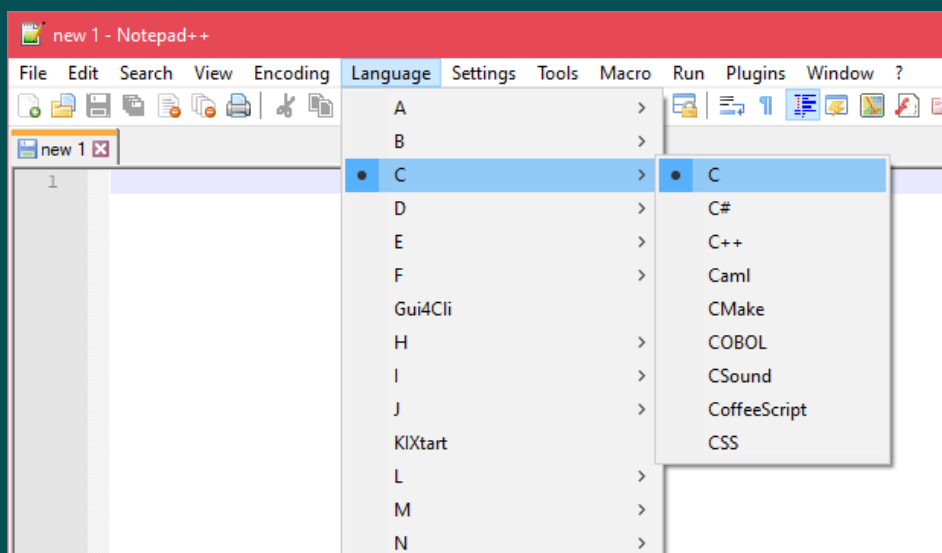
PERSIAPAN

Sebelum ngoding, ada beberapa software yang harus dipersiapkan yaitu **text editor** dan **compiler**.

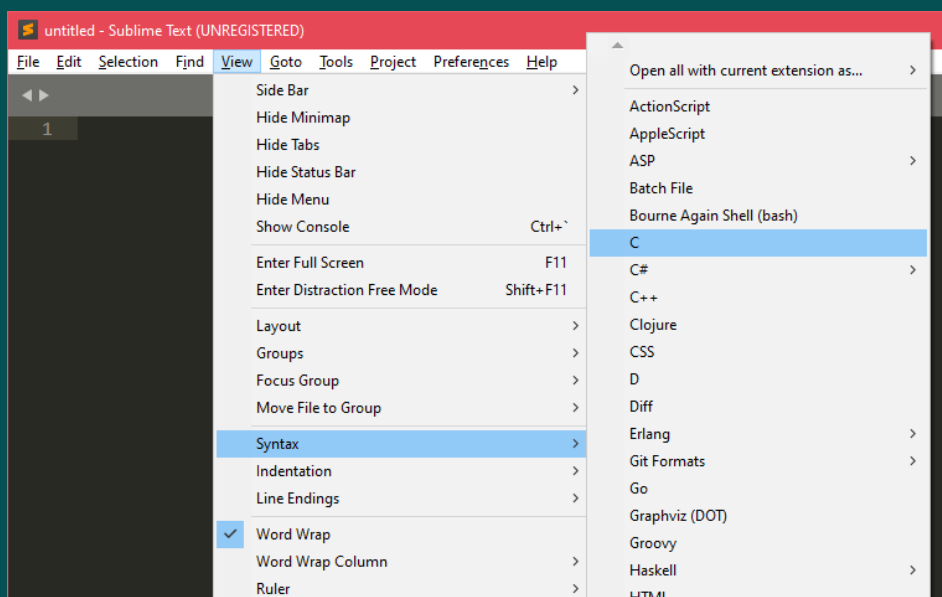
Text Editor

Text editor adalah software untuk **menuliskan kode program**. Disarankan menggunakan text editor yang mendukung bahasa C seperti Notepad++, Sublime Text atau VS Code. Kemudian atur text untuk meng-highlight sintaksis bahasa C.

Disarankan tidak memakai Integrated Development Environment (IDE) ketika mempelajari sebuah bahasa.



Setting pada notepad++



Setting pada sublime



Compiler

Compiler adalah sebuah program komputer yang akan **mengubah sebuah instruksi ke dalam bahasamesin**. (Cambridge University Press, 2013). Dalam hal ini diubah ke dalam bentuk bahasa Assembly sehingga selanjutnya akan menghasilkan suatu file execute (.exe), di mana file tersebut dapat berdiri sendiri tanpa memerlukan perangkat lunak lainnya.

Salah satu compiler yang digunakan untuk mengkompilasi kode program berbahasa C adalah GNU Compiler Collective (GCC).

Windows

Cara menginstal salah satu compiler yaitu MinGW.

1. Instalasi MinGW

Instalasi disarankan menggunakan installer yang akan mendownload kembali paket MinGW yang diperlukan dan otomatis mengatur sistem agar bisa langsung digunakan.

2. Mengatur Path

- a. Copy alamat "C:\MinGW\bin"
- b. Buka Control Panel, lalu buka bagian System,
- c. Pilih menu "Advancement system settings"
- d. Pilih menu "Environment Variables"
- e. Pilih "Path" di "System Variables"
- f. Pilih "Edit" ...
- g. Pilih "New", lalu masukan alamat "C:\MinGW\bin". Lalu pilih "Ok"

MENULIS PROGRAM DALAM BAHASA C

Setelah program komputer yang dibutuhkan sudah dipersiapkan. Selanjutnya kita mulai ngoding.

Template Program Bahasa C

Berikut adalah template untuk program bahasa C.

```
HelloWorld.c x
1  #include<stdio.h> //ini adalah header
2
3  int main(){
4
5      //tuliskan kode program
6
7      return 0;
8  }
9
```

Menulis Kode Program Menampilkan Hello World

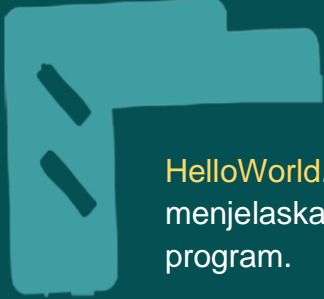
Mari kembangkan kodingan di atas menjadi program yang akan menampilkan kata "Hello World ^_^". Caranya dengan menambahkan kode pada baris ke-6, seperti gambar berikut:

```
HelloWorld.c x
1  #include<stdio.h> //ini adalah header
2
3  int main(){
4
5      //menampilkan Hello World ke layar
6      printf("Hello World ^_^\n");
7
8      return 0;
9  }
```

Masih bingung dengan maksud kode di atas? Berikut adalah gambaran singkat kodingan diatas.

HelloWorld.c:1 `#include`, cara untuk mengimpor konten dari sebuah file ke source code. Pada Bahasa C ekstensi `.h` digunakan untuk file header yang berisi fungsi untuk digunakan pada program anda.

HelloWorld.c:3 Awal dari fungsi utama (Main). Untuk memulai bagian isi dari fungsi, digunakan karakter Opening Brace " `{` " yang mengindikasikan awal dari sebuah blok. Kode program yang akan dijalankan harus ditulis setelah karakter tersebut.



HelloWorld.c:5 Ini merupakan komentar. Komentar biasanya digunakan untuk menjelaskan source code. Komentar tidak akan mengganggu jalan kerjanya sebuah program.

HelloWorld.c:6 Memanggil fungsi printf untuk menampilkan tulisan ke layar, dengan isi teks “Hello World ^_^\\n” dimana kata yang akan ditampilkan berada di antara tanda dua kutip. Sedangkan \\n berfungsi untuk memberi baris baru (newline/enter). Jangan lupa karakter semicolon atau titik koma untuk mengakhiri pemanggilan fungsi.

HelloWorld.c:8 Return dari fungsi main yang memberikan sistem operasi tanda untuk keluar dari program setelah kode dijalankan.

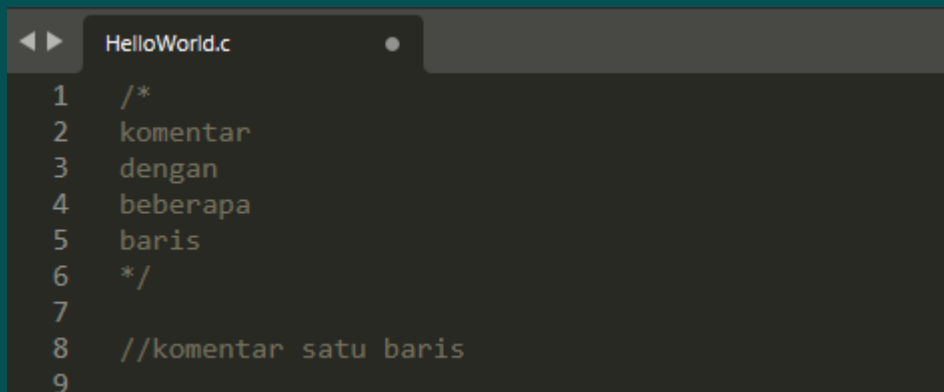
HelloWorld.c:9 Pada akhir kode, Main ditutup dengan karakter closing brace “ } “ dan menjadi akhir dari program.

Komentar

Komentar merupakan bagian **kode program yang tidak dieksekusi** oleh kompiler atau interpreter saat program dijalankan. Komentar digunakan untuk menjelaskan kode, mengingat atau memahami alur program yang telah dibuat.

Penulisan komentar untuk satu baris menggunakan double forward slash (//), sedangkan penulisan komentar lebih dari satu baris diawali dengan /* dan diakhiri dengan */.

Contoh :



```
1  /*
2  komentar
3  dengan
4  beberapa
5  baris
6  */
7
8  //komentar satu baris
9
```

KOMPILASI

Kompilasi adalah aktivitas menerjemahkan bahasa program ke bahasa mesin. Kompilasi dilakukan agar kode program yang sudah ditulis menjadi sebuah program.

Misal kode diatas di simpan pada **Drive D** pada **folder Praktikum**.

```
C:\ Select Command Prompt
Microsoft Windows [Version 10.0.19043.1165]
(c) Microsoft Corporation. All rights reserved.

D:\Praktikum>gcc HelloWorld.c -o HelloWorld.exe

D:\Praktikum>HelloWorld
Hello World ^_^
```

Biasanya CMD awalnya menampilkan drive C. Untuk berpindah dari drive C ke drive D, gunakan perintah "D:". Untuk mengubah direktori/folder, gunakan perintah cd. Nama folder harus sama persis. Contoh:

```
C:\ Select Command Prompt
Microsoft Windows [Version 10.0.19043.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS>d:

D:\>cd praktikum

D:\Praktikum>gcc HelloWorld.c -o HelloWorld.exe

D:\Praktikum>HelloWorld
Hello World ^_^
```

Untuk mengompilasi gunakan sintaksis berikut:

```
gcc [nama_source_kode] -o [nama_program_yang_dihasilkan]
gcc HelloWorld.c -o HelloWorld.exe
```

Jalankan program dengan mengetik nama programnya.
helloWorld.exe

Untuk macOS dan linux jalankan programnya dengan
./helloworld.exe

VARIABEL DAN TIPE DATA

Pada bagian ini kita akan membahas mengenai variabel dan tipe data.

Variabel

Variabel merupakan tempat untuk **menyimpan/menampung suatu nilai**, dan nilai yang ditampung dapat diubah selama eksekusi program berlangsung. Variabel lebih dikenal sebagai *identifier*.

Aturan penamaan variabel:

- **Case-sensitive**. Dimana dapat membedakan antara variabel yang ditulis huruf besar dengan variabel yang ditulis huruf kecil. Contoh: aku berbeda dengan Aku.
- Nama variabel **tidak boleh hanya** berupa **angka** ataupun **diawali** dengan **angka**.
- Nama variabel **tidak boleh** mengandung **spasi**. Sebagai **gantinya** dapat dengan menggunakan karakter **underscore** (`_`) atau dengan menggabungkan kata ke bentuk **camelCase**.
- Nama variabel **tidak boleh** mengandung karakter-karakter **simbol** (`@`, `#`, `$`, `%`, dll.) walaupun terletak di tengah ataupun akhir nama variabel.
- Nama variabel **tidak boleh** menggunakan **kata kunci** maupun **makro** yang telah didefinisikan dalam Bahasa C, contoh: **void**, **printf**, **scanf**, **return**, dll.
- Penamaan variabel diharapkan merupakan **deskripsi** dari penggunaannya. Sebagai contoh, variabel menampung nilai luas diberi nama luas.

TIPE DATA

Tipe data adalah jenis data berdasarkan **isi dan sifatnya**. Dalam logika kita tipe data adalah jenis dari sesuatu yang dapat dimasukkan ke dalam kotak kosong yang hanya khusus dibuat untuk jenis benda dengan jenis tertentu.

a. Bilangan Bulat (Integer)

| Tipe Data | Jangkauan | Format Sintaksis |
|-----------|--------------------------------|------------------|
| Int | -32768, ... , +32768 | %d |
| long int | -2147483648, ... , +2147483647 | %l |
| short int | -128, ... , +127 | %d |

b. Bilangan Rill (Floating-Point)

| Tipe Data | Jangkauan | Format Sintaksis |
|-------------|----------------------------|------------------|
| float | 1.2e-38 sampai 3.4e+38 | %f |
| double | 2.3e-308 sampai 1.7e+308 | %f |
| long double | 3.4e-4932 sampai 1.7e+4932 | %lf |

c. Karakter (Character)

| Tipe Data | Jangkauan | Format Sintaksis |
|-------------|------------------------------------|------------------|
| char | -128 sampai +127 atau 0 sampai 255 | %c |
| signed char | -128 sampai +127 | %c |

Tipe data ini mempunyai jangkauan karakter ASCII ke 0 sampai karakter ASCII 255. Tipe data ini bisa ditampilkan dengan suatu karakter atau suatu bilangan. Tetapi jika ingin diisi dengan angka maka angka tersebut harus diberi tanda kutip. Jika ingin memanggil tipe data karakter di dalam fungsi printf harus mencantumkan %c. Apabila dipanggil dengan format &d atau %i maka yang tampil adalah karakter ASCII dari data tersebut.

Deklarasi Variabel

Untuk mendeklarasikan variabel dalam bahasa C, digunakan sintaksis berikut:
[tipe_data] [nama_variabel];

```
int main(){  
  
    //contoh eklarasi variabel beberapa baris  
    int number;  
    double rasional;  
    char karakter;  
  
    //contoh deklarasi satu baris  
    int x, y;  
    /*  
    sama saja dengan  
    int x;  
    int y;  
    */  
  
    return 0;  
}
```

OPERATOR

Operator adalah **simbol atau tanda** yang diletakkan pada **dua buah operan** sehingga dapat menghasilkan sebuah **hasil**. Operator dibagi menjadi berbagai jenis, diantaranya:

Operator Aritmatika

Operator yang digunakan untuk melakukan **operasi matematik** dengan **dua operand**.

| Operator | Jenis Operator | Contoh |
|----------|----------------------|--------|
| + | Penjumlahan | 1 + 1 |
| - | Pengurangan | 11 - 1 |
| * | Perkalian | 3 * 6 |
| / | Pembagian | 32 / 8 |
| % | Modulus (siswa bagi) | 21 % 2 |

Modulus

1. $15 \% 3 = ?$

2. $78 \% 5 = ?$

3. $170 \% 13 = ?$

Operator Assignment

Operator penugasan atau operator assignment (=) digunakan untuk **memasukkan nilai** ke dalam suatu **variabel**.

| Operator | Contoh | Artinya |
|----------|--------|-----------|
| = | a = b | a = b |
| += | a += b | a = a + b |
| -= | a -= b | a = a - b |
| *= | a *= b | a = a * b |
| /= | a /= b | a = a / b |
| %= | a %= b | a = a % b |

```
int main(){  
  
    //deklarasi dan pemasukan nilai dalam satu baris  
    char alpabet = 'n'; //assignment tipe data char menggunakan tanda petik (' ')  
  
    float number; //deklarasi variabel  
    number = 6.6; //pemasukan nilai ke variabel "number"  
  
    //deklarasi beberapa variabel dan pemasukan nilai sekaligus  
    float num1, num2, num3;  
    num1 = num2 = num3 = 6; //kedua variabel akan berisi nilai 6.6  
  
    num1 += num2; //num1 akan berisi nilai 12  
    num1 -= num2; //num1 akan berisi nilai 6  
    num1 *= num2; //num1 akan berisi nilai 36  
    num1 /= num2; //num1 akan berisi nilai 6  
    num1 %= num2; //num1 akan berisi nilai 0  
  
    return 0;  
}
```

Operasi Unary

Operator yang digunakan untuk melakukan **operasi matematik** yang hanya menggunakan **satu operand**.

| Operator | Jenis Operasi | Contoh |
|----------|---------------------------------|--------|
| + | Membuat nilai positif | +10 |
| - | Membuat nilai negatif | -10 |
| ++ | Increment (menambahkan nilai 1) | a++ |
| -- | Decrement (mengurangi nilai 1) | a-- |

MEKANISME INPUT DAN OUTPUT

Salah satu cara untuk melakukan input dan output dalam Bahasa pemrograman C ialah dengan menggunakan `printf` dan `scanf`. Masih banyak cara lainnya tidak dibahas di modul ini. `Printf` dan `scanf` merupakan fungsi dari library `stdio.h`. Pastikan anda melakukan include di awal source code.

Formatted Print

Formatted print atau pencetakan yang diformat, merupakan fungsi untuk menampilkan data ke output standard (layar). Penulisannya sebagai berikut:

```
printf("konten yang akan di print");
```

Untuk menampilkan data dari variabel direpresentasikan dengan simbol formatnya kemudian nama variabel disebutkan setelahnya. Sehingga ditulis sebagai berikut:

```
printf("simbol format", nama variabel);
```

Contoh:

```
float number = 6.6;
```

```
printf("%f\n", number);
```

```
//      ^      ^ menyebutkan nama variabel yang akan di print isinya
```

```
//      ^ perwakilan isi variabel dengan simbol format
```

```
//menampilkan 2 angka dibelakang koma
```

```
printf("%0.2f\n", number);
```

```
//deklarasi variabel
```

```
int angkatan = 2021;
```

```
char kelas = 'C';
```

```
//menampilkan "Saya kelas C angkatan 2020"
```

```
printf("Saya kelas %c angkatan %d", kelas, angkatan);
```

```
int main(){

    float number = 6.6;
    printf("%f\n", number);
    //      ^      ^ menyebutkan nama variabel yang akan di print isinya
    //      ^ perwakilan variabel dengan simbol format

    //menampilkan 2 angka dibelakang koma
    printf("%0.2f\n", number);

    //deklarasi variabel
    int angkatan = 2021;
    char kelas = 'C';

    //menampilkan "Saya kelas A angkatan 2021"
    printf("Saya kelas %c angkatan %d\n", kelas, angkatan);

    return 0;
}
```

Formatted Scan

Formatted scan atau pemindaian terformat, merupakan fungsi untuk meminta masukan (input) dari user melalui input standar (console).

Untuk meminta masukan dari user, variabel diwakilkan dengan simbol formatnya kemudian menyebutkan nama variabelnya. Sebelum nama variabel harus menggunakan simbol "&". Sehingga penulisannya seperti berikut:

`scanf ("simbol format", &nama variabel);`

Khusus untuk karakter, sebelum simbol format diberi 1 spasi.

Contoh:

```
char alpabet; //deklarasi variabel
```

```
scanf(" %c\n", &alpabet); //meminta masukan user sebuah alpabet
```

```
//      ^      ^ menyebutkan nama variabel yang akan di masukan nilai
```

```
//      ^ perwakilan variabel dengan simbol format
```

```
int number1; //deklarasi variabel
```

```
float number2; //deklarasi variabel
```

```
//meminta masukan dua kali untuk variabel number1 dan number2
```

```
scanf("%d %f", &number1, &number2);
```

```
//menampilkan semua variabel
```

```
printf("%c %d %.2f\n", alpabet, number1, number2);
```

```
int main(){

    char alpabet;    //deklarasi variabel
    scanf(" %c\n", &alpabet); //meminta masukan user sebuah alpabet
    //      ^      ^ menyebutkan nama variabel yang akan di masukan nilai
    //      ^ perwakilan variabel dengan simbol format

    int number1;    //deklarasi variabel
    float number2;  //deklarasi variabel

    //meminta masukan dua kali untuk variabel number1 dan number2
    scanf("%d %f", &number1, &number2);

    //menampilkan semua variabel
    printf("%c %d %.2f\n", alpabet, number1, number2);

    return 0;
}
```

MENULIS KALKULATOR SEDERHANA

```
MiniKalkulator.c x
1  #include <stdio.h>
2
3  int main(){
4
5      //deklarasi variabel penampung num1, num2 dan hasil operasi
6      int num1, num2;
7      int tambah, kurang, kali, bagi, mod, jumKuadrat, kuadratJum;
8
9      //meminta user untuk memasukan num1
10     printf("Masukan angka pertama: ");
11     scanf("%d", &num1);
12
13     //meminta user untuk memasukan num1
14     printf("Masukan angka kedua: ");
15     scanf("%d", &num2);
16
17     //proses operasi
18     tambah = num1 + num2;
19     kurang = num1 - num2;
20     kali = num1 * num2;
21     bagi = num1 / num2;
22     mod = num1 % num2;
23     jumKuadrat = (num1 * num1) + (num2 * num2);
24     kuadratJum = tambah * tambah;
25
26     //menampilkan hasil operasi
27     printf("Penjumlahan      : %d\n", tambah);
28     printf("Pengurangan      : %d\n", kurang);
29     printf("Perkalian         : %d\n", kali);
30     printf("Pembagian         : %d\n", bagi);
31     printf("Modulus           : %d\n", mod);
32     printf("Jumlah Kuadrat    : %d\n", jumKuadrat);
33     printf("Kuadrat Jumlah    : %d\n", kuadratJum);
34
35     return 0;
36 }
```

LATIHAN SOAL

1. Seorang anak SD memakan kue berbentuk **tabung** yang diameternya **d cm** dan tebalnya berukuran **t cm**. Ia ingin menghitung **volume** tersebut tersebut, tetapi ia lupa caranya. Bantulah dia.

- Masukan: Sebuah baris berisi dua buah bilangan bulat d dan t.
- Keluaran: Sebuah baris berisi sebuah bilangan riil yang menyatakan volume dari kue tersebut dalam cm kubik, dengan tepat dua angka di belakang koma.

Contoh Masukan 1

Diameter : 3

Tinggi : 4

Contoh Masukan 2

Diameter : 123

Tinggi : 456

Contoh Keluaran 1

Volumenya: 28.29 cm kubik

Contoh Keluaran 2

Volumenya: 5420504.50 cm kubik

2. Bibi May memiliki **N kue**. Ia ingin **membagi-bagikan** kue-kue tersebut kepada **M orang** keponakannya dengan sama banyak. Bibi May juga menyadari bahwa bisa saja terdapat sisa kue karena banyaknya kue **tidak habis dibagi** banyaknya keponakannya. Bantulah Bibi May untuk menentukan berapa banyak kue yang harus dia berikan kepada masing-masing keponakannya, dan berapa sisanya.

- Masukan: Sebuah baris berisi dua buah bilangan bulat N dan M.
- Keluaran: Baris pertama berisi masing-masing A, dengan A adalah banyaknya light stick yang diberikan kepada masing-masing temannya. Baris kedua berisi bersisa B, dengan B adalah banyaknya sisa light stick Lisa.

Contoh Masukan 1

Jumlah kue: 100

Jumlah keponakan :7

Contoh Masukan 2

Jumlah kue : 1010

Jumlah keponakan :11

Contoh Keluaran 1

Masing-Masing: 14 buah

Sisa Light Stick: 2 buah

Contoh Keluaran 2

Masing-Masing: 91 buah

Sisa Ligh Stick: 9 buah



PENUTUP

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua elemen yang telah mendukung berjalannya sesi praktikum pada mata kuliah Algoritma dan Pemrograman I ini. Semoga apa yang telah kita laksanakan dapat menjadi berkah dan memberikan manfaat bagi kita semua dikemudian hari.

DAFTAR PUSTAKA

- Tim Asisten Praktikum Algoritma dan Pemrograman I Angkatan 11. (2020). *Pengenalan Variabel, Operasi & Mekanisme I/O*. Modul Praktikum Algoritma dan Pemrograman I. Bandung: Universitas Pendidikan Indonesia.
- Bachtiar, Adam Mukharil. (2016). *Pemrograman C dan C++*. Bandung: Informatika Bandung.