

**LAPORAN AKHIR PRAKTIKUM**  
**MATA KULIAH**  
**MODUL 6**  
**FUNGSI DAN PROSEDUR**



**Nama** : Kurnia Fajar Rahyudi Putra  
**No. BP** : 2211512013  
**Hari/Tanggal** : Kamis/ 8 Juni 2023  
**Shift** : 1

**Dosen** : Dodon Yendri, M.Kom

**LABORATORIUM KOMPUTER DAN JARINGAN**  
**DEPARTEMEN TEKNIK KOMPUTER**  
**FAKULTAS TEKNOLOGI INFORMASI**  
**UNIVERSITAS ANDALAS**  
**PADANG**  
**2023**

# **BAB I**

## **PENDAHULUAN**

### 1.1 Tujuan

- Memperkenalkan kepada mahasiswa konsep pemrograman modular
- Memperkenalkan kepada mahasiswa pemakaian fungsi/function
- Memperkenalkan kepada mahasiswa pemakaian prosedur
- Mempraktekkan pemakaian fungsi
- Mempraktekkan pemakaian prosedur

### 1.2 Landasan Teori (min 3 hal di luar gambar dan tujuan)

- **Definisi Modular**

Pemrograman Modular adalah suatu teknik pemrograman di mana program yang biasanya cukup besar dibagi-bagi menjadi beberapa bagian program yang lebih kecil sehingga akan mudah dipahami dan dapat digunakan kembali, baik untuk program itu sendiri maupun program lain yang memiliki proses yang sama.

- **Pengertian Fungsi**

**Fungsi (Function)** merupakan blok dari kode yang dirancang untuk melaksanakan tugas khusus. Kegunaan dari fungsi ini adalah untuk:

1. Mengurangi pengulangan penulisan program yang berulang atau sama.
2. Program menjadi lebih terstruktur, sehingga mudah dipahami dan dapat lebih dikembangkan.

Fungsi-fungsi yang sudah kita kenal sebelumnya adalah fungsi `main()`, yang bersifat mutlak, karena fungsi ini program akan dimulai, sebagai contoh yang lainnya fungsi `printf()`, `cout()` yang mempunyai tugas untuk menampilkan informasi atau data kelayar dan masih banyak lainnya.

Terdapat lima konsep dasar mengenai fungsi dalam pemrograman, yaitu:

1. Fungsi adalah objek (bagian program/rutin) yang mengerjakan suatu tugas tertentu dan digunakan untuk memodularkan program dengan

suatu ciri mengembalikan suatu nilai (return value)

2. Fungsi dapat digunakan untuk menghindari penulisan yang sama yang ditulis berulang- ulang.
3. Semua variabel yang dideklarasikan dalam fungsi merupakan variabel lokal, yang hanya diketahui dalam fungsi bersangkutan
4. Fungsi bisa memiliki parameter yang menyediakan komunikasi antara fungsi dengan bagian yang memanggil fungsi tersebut (Parameter formal dan Parameter aktual).
5. Parameter bersifat lokal.

- **Bentuk Umum Fungsi**

Sebuah fungsi sederhana mempunyai bentuk penulisan sebagai berikut :

```
nama_fungsi(argumen1, argumen2, ...)  
{  
... pernyataan / perintah;  
... pernyataan / perintah;  
... pernyataan / perintah;  
}
```

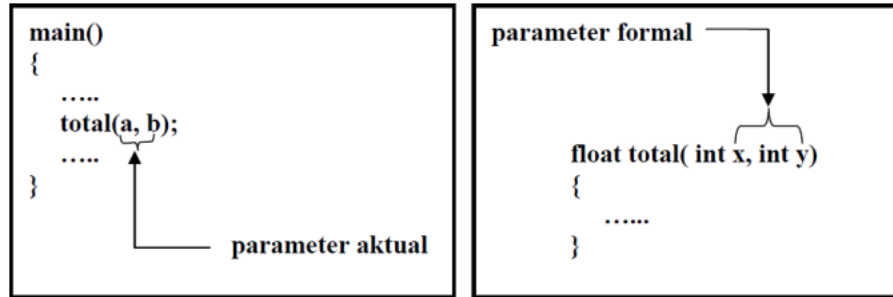
Keterangan:

1. Nama fungsi, boleh dituliskan secara bebas dengan ketentuan, tidak menggunakan spasi dan nama-nama fungsi yang mempunyai arti sendiri.
2. Argumen, diletakan diantara tanda kurung “( )” yang terletak dibelakang nama fungsi.
3. Argumen boleh diisi dengan suatu data atau dibiarkan kosong. Pernyataan / perintah, diletakan diantara tanda kurung ‘{ }’.

- **Parameter Fungsi**

Terdapat dua macam para parameter fungsi, yaitu :

1. **Parameter formal** adalah variabel yang terdapat pada daftar parameter yang berada didalam definisi fungsi.
2. **Parameter Aktual** adalah variabel yang digunakan pada pemanggilan suatu fungsi.



- **Fungsi Rekursif**

Fungsi rekursif adalah suatu fungsi yang memanggil dirinya sendiri, artinya fungsi tersebut dipanggil di dalam tubuh fungsi itu sendiri. Fungsi rekursif sangat berguna bila diimplementasikan untuk pekerjaan pengurutan data, atau menghitung nilai factorial suatu bilangan.

```
return_data_type function_name(parameter_list)
{
    ...
    function_name(...);
    ...
}
```

- **Definisi Prosedur**

Prosedur sama dengan fungsi prosedur juga merupakan bagian dari sub program yang mengerjakan suatu perintah/tugas tertentu. Prosedur bermanfaat saat kita ingin melakukan perintah yang sama berulang-ulang kali, kita hanya perlu membuat sekali dan nantinya dapat dipergunakan dengan cara dipanggil sesuai dengan kebutuhan.

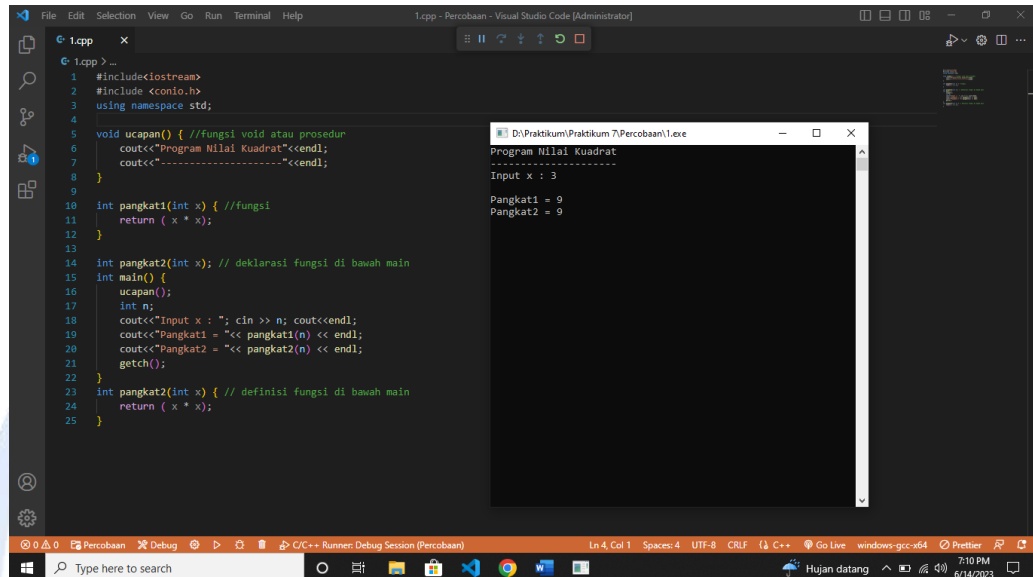
- **Perbedaan Fungsi dan Prosedur**

Pada dasarnya tidak ada perbedaan yang begitu signifikan antar prosedur dan fungsi keduanya merupakan sebuah sub program yang dapat mengerjakan tugas tertentu, namun jika di cermati fungsi dideklarasikan dengan menggunakan tipe data misalnya int penjumlahan(), fungsi tersebut wajib harus menghasilkan nilai balik (return value). Sementara untuk prosedur di deklarasikan biasanya dengan tipe void tipe void adalah salah satu jenis tipe data yang tidak berarti apa-apa (Valueless) tipe void biasanya digunakan untuk prosedur yang pada dasarnya tidak mempunyai nilai balik. Karena tipe void bersifat valueless maka pada saat pendeklarasian prosedur tidak menyebutkan tipe void pun tidak apa-apa

## BAB II

### PERCOBAAN

#### 2.1 Percobaan 6.1



The screenshot shows the Visual Studio Code editor with a C++ file named `1.cpp`. The code defines two functions, `ucapan()` and `pangkat1(int x)`, and a `main()` function. The `ucapan()` function prints "Program Nilai Kuadrat" and a separator line. The `pangkat1` function calculates the square of its argument. The `main` function prompts the user for input, calls `pangkat1` with the input, and prints the result. The output window shows the program's execution: "Program Nilai Kuadrat", a separator line, "Input x : 3", and the calculated squares "Pangkat1 = 9" and "Pangkat2 = 9".

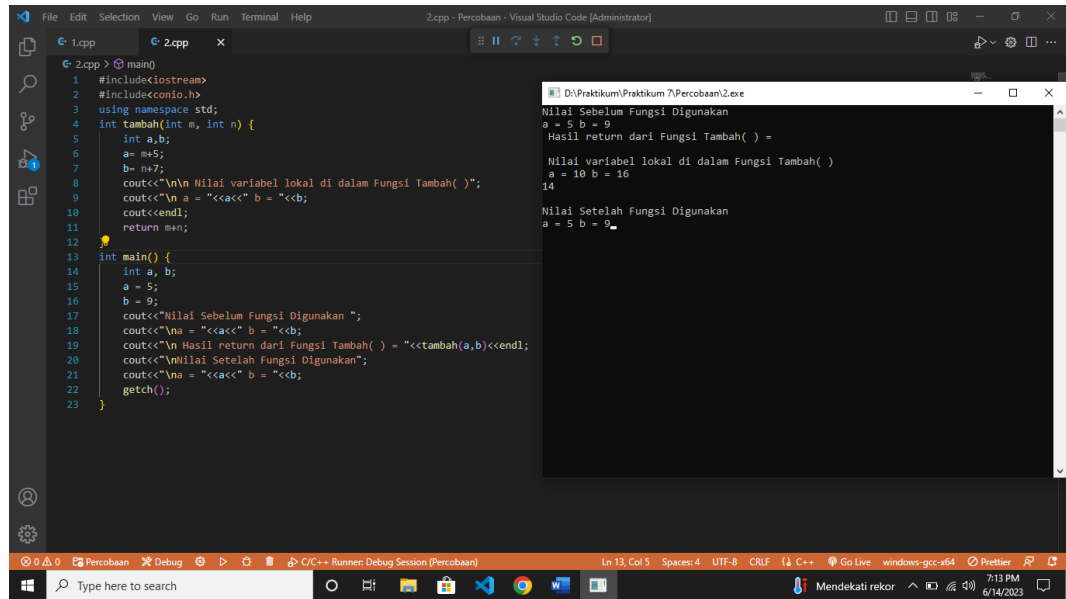
```
1 #include<iostream>
2 #include <conio.h>
3 using namespace std;
4
5 void ucapan() { //fungsi void atau prosedur
6     cout<<"Program Nilai Kuadrat"<<endl;
7     cout<<"-----"<<endl;
8 }
9
10 int pangkat1(int x) { //fungsi
11     return ( x * x);
12 }
13
14 int pangkat2(int x); // deklarasi fungsi di bawah main
15 int main() {
16     ucapan();
17     int n;
18     cout<<"Input x : "; cin >> n; cout<<endl;
19     cout<<"Pangkat1 = "<< pangkat1(n) << endl;
20     cout<<"Pangkat2 = "<< pangkat2(n) << endl;
21     getch();
22 }
23 int pangkat2(int x) { // definisi fungsi di bawah main
24     return ( x * x);
25 }
```

Output:

```
Program Nilai Kuadrat
-----
Input x : 3
Pangkat1 = 9
Pangkat2 = 9
```

Program di atas menggunakan penulisan model fungsi karena program membutuhkan input. Oleh karena itu terdapat parameter yang berfungsi untuk menyimpan nilai yang akan diinputkan ke dalam fungsi. Tipe data yang digunakan pada program ini adalah integer. Parameter ini akan menyimpan nilai yang diinputkan ke fungsi ucapan. Program di atas dibuat untuk menentukan nilai kuadrat dari angka yang diinputkan.

## 2.2 Percobaan 6.2



The screenshot shows a Visual Studio Code editor with a C++ file named `2.cpp`. The code defines a function `tambah` and a `main` function. The `main` function calls `tambah` and prints the values of `a` and `b` before and after the function call. The output window shows the execution results.

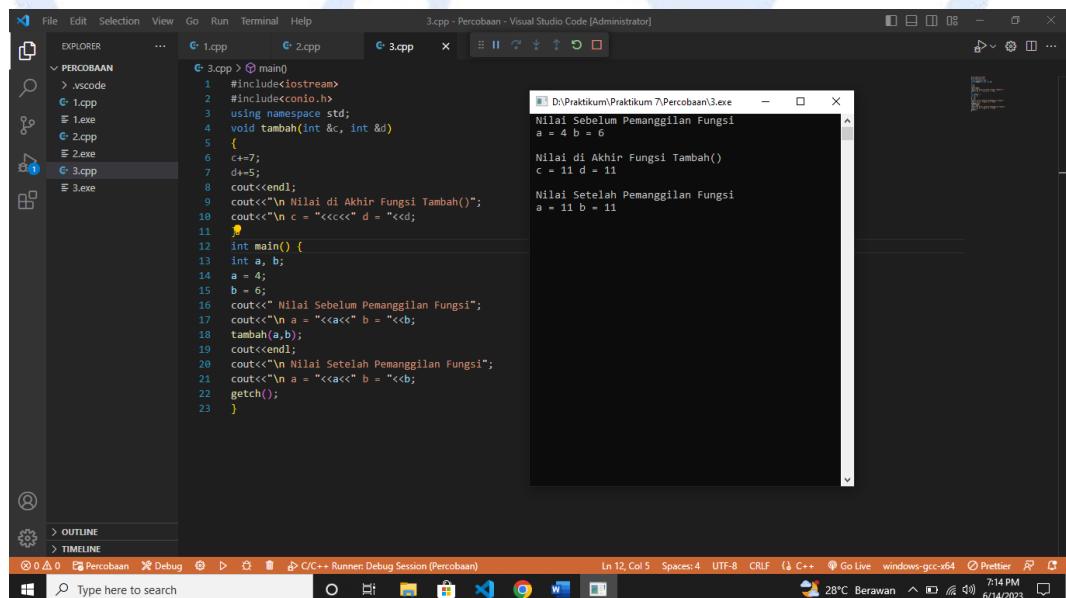
```
1 #include<iostream>
2 #include<conio.h>
3 using namespace std;
4 int tambah(int m, int n) {
5     int a,b;
6     a= m+5;
7     b= n+7;
8     cout<<"\n\n Nilai variabel lokal di dalam Fungsi Tambah( )";
9     cout<<"\n a = "<<a<<" b = "<<b;
10    cout<<endl;
11    return m+n;
12 }
13 int main() {
14     int a, b;
15     a = 5;
16     b = 9;
17     cout<<"Nilai Sebelum Fungsi Digunakan ";
18     cout<<"\n a = "<<a<<" b = "<<b;
19     cout<<"\n Hasil return dari Fungsi Tambah( ) = "<<tambah(a,b)<<endl;
20     cout<<"\nNilai Setelah Fungsi Digunakan";
21     cout<<"\n a = "<<a<<" b = "<<b;
22     getch();
23 }
```

Output:

```
Nilai Sebelum Fungsi Digunakan
a = 5 b = 9
Hasil return dari Fungsi Tambah( ) =
Nilai variabel lokal di dalam Fungsi Tambah( )
a = 10 b = 16
14
Nilai Setelah Fungsi Digunakan
a = 5 b = 9
```

Program di atas menggunakan operasi fungsi dengan pemanggilan nilai. Pada program di atas fungsi didefinisikan langsung. Oleh karena itu fungsinya diletakkan di atas fungsi main. Jika fungsi diletakkan di bawah fungsi maka, program akan error. Tipe data yang digunakan pada program ini adalah integer. Perubahan nilai di fungsi (parameter formal) tidak akan mengubah nilai asli di bagian program yang memanggilnya.

## 2.3 Percobaan 6.3



The screenshot shows a Visual Studio Code editor with a C++ file named `3.cpp`. The code defines a function `tambah` and a `main` function. The `main` function calls `tambah` and prints the values of `a` and `b` before and after the function call. The output window shows the execution results.

```
1 #include<iostream>
2 #include<conio.h>
3 using namespace std;
4 void tambah(int &c, int &d)
5 {
6     c+=7;
7     d+=5;
8     cout<<endl;
9     cout<<"\n Nilai di Akhir Fungsi Tambah()";
10    cout<<"\n c = "<<c<<" d = "<<d;
11 }
12 int main() {
13     int a, b;
14     a = 4;
15     b = 6;
16     cout<<" Nilai Sebelum Pemanggilan Fungsi";
17     cout<<"\n a = "<<a<<" b = "<<b;
18     tambah(a,b);
19     cout<<endl;
20     cout<<"\n Nilai Setelah Pemanggilan Fungsi";
21     cout<<"\n a = "<<a<<" b = "<<b;
22     getch();
23 }
```

Output:

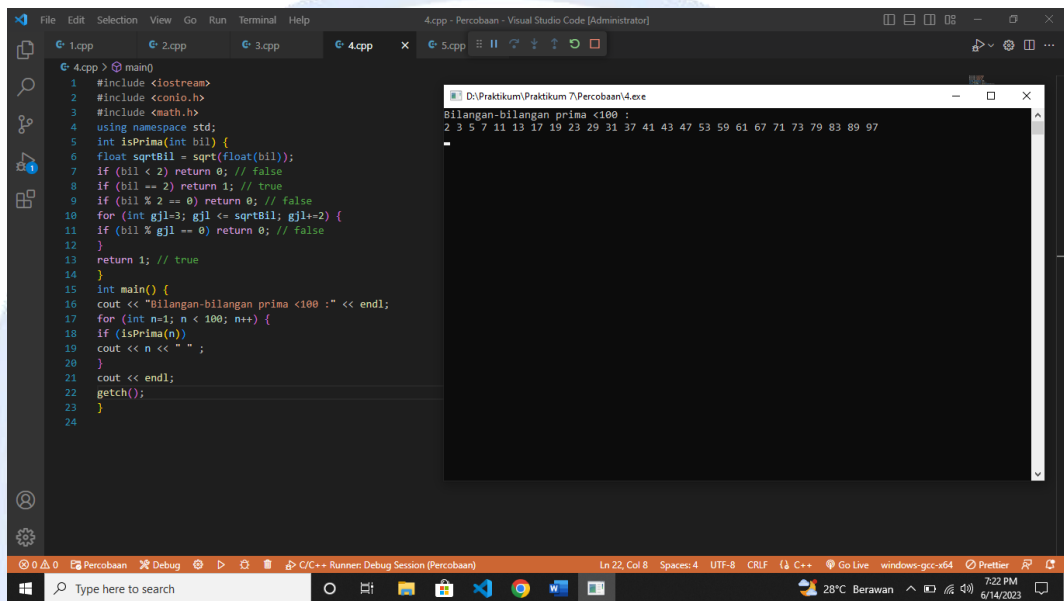
```
Nilai Sebelum Pemanggilan Fungsi
a = 4 b = 6

Nilai di Akhir Fungsi Tambah()
c = 11 d = 11

Nilai Setelah Pemanggilan Fungsi
a = 11 b = 11
```

Program di atas menggunakan operasi fungsi pemanggilan dengan acuan. Pemanggilan dengan acuan merupakan upaya melewati alamat dari suatu variabel ke dalam fungsi. Alamat yang dikirimkan ke fungsi adalah alamat letak dari nilai datanya, bukan nilai datanya. Tipe data yang digunakan dalam program ini adalah integer. Perubahan nilai pada fungsi akan mengubah nilai asli di bagian program yang memanggil fungsi.

## 2.4 Percobaan 6.4



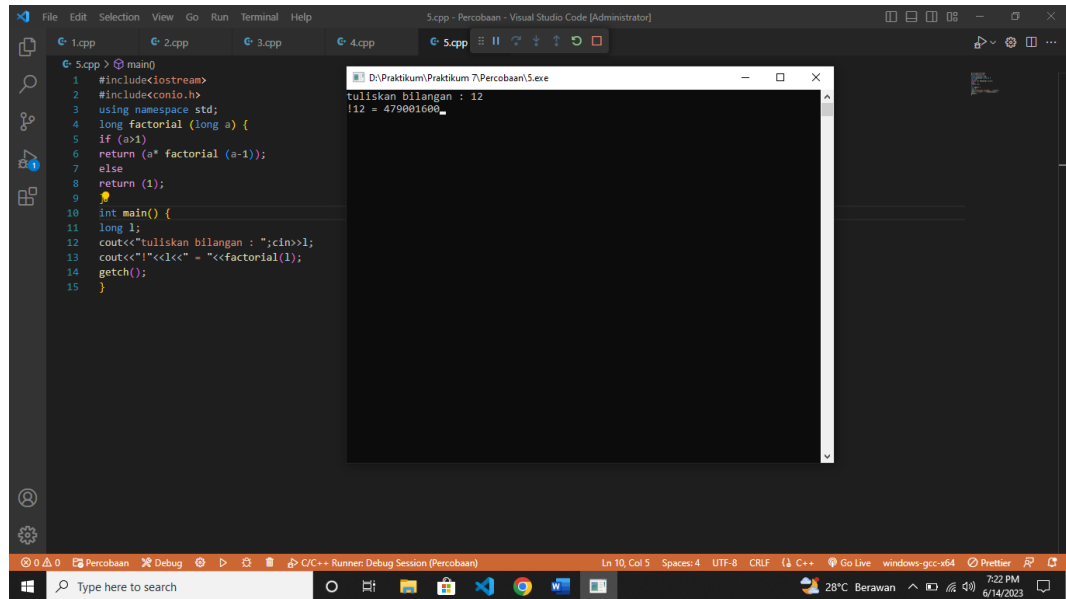
```
1 #include <iostream>
2 #include <conio.h>
3 #include <math.h>
4 using namespace std;
5 int isPrima(int bil) {
6     float sqrtBil = sqrt(float(bil));
7     if (bil < 2) return 0; // false
8     if (bil == 2) return 1; // true
9     if (bil % 2 == 0) return 0; // false
10    for (int gjl=3; gjl <= sqrtBil; gjl+=2) {
11        if (bil % gjl == 0) return 0; // false
12    }
13    return 1; // true
14 }
15 int main() {
16     cout << "Bilangan-bilangan prima <100 : " << endl;
17     for (int n=1; n < 100; n++) {
18         if (isPrima(n)) {
19             cout << n << " ";
20         }
21     }
22     cout << endl;
23     getch();
24 }
```

D:\Praktikum\Praktikum 7\percobaan\4.exe  
Bilangan-bilangan prima <100 :  
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

Program di atas menggunakan operasi fungsi boolean. Tipe data boolean adalah tipe data yang hanya bisa diisi salah satu dari 2 nilai: true atau false. Oleh karena itu, pada program di atas digunakan kondisi percabangan seperti if. Tipe data yang digunakan pada program di atas adalah integer. Fungsi boolean pada program ini digunakan untuk menentukan jumlah bilangan prima pada <100.

## 2.5 Percobaan 6.5





```
5.cpp > main()
1 #include<iostream>
2 #include<conio.h>
3 using namespace std;
4 long factorial (long a) {
5     if (a>1)
6         return (a* factorial (a-1));
7     else
8         return (1);
9 }
10 int main() {
11     long l;
12     cout<<"tuliskan bilangan : ";cin>>l;
13     cout<<"!"<<l<<" = "<<factorial(l);
14     getch();
15 }
```

D:\Praktikum\Praktikum 7\percobaan\5.exe

tuliskan bilangan : 12  
!12 = 479001600

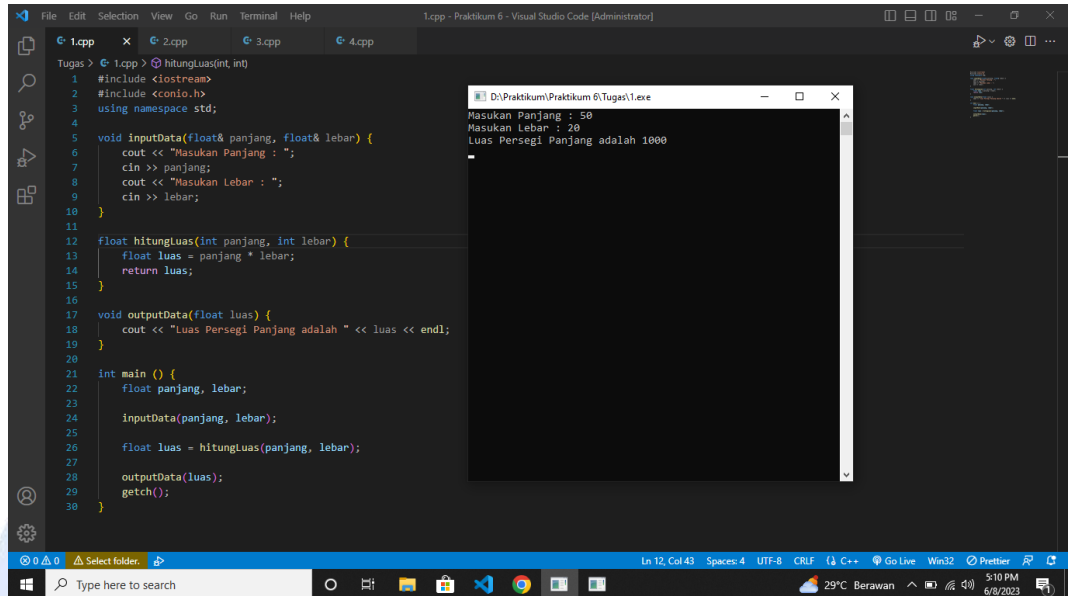
Program di atas menggunakan operasi fungsi rekursif. Fungsi rekursif ini merupakan fungsi yang memanggil diri. Tipe data yang digunakan pada program ini adalah integer. Program di atas merupakan program untuk mencari faktorial dari nilai yang diinputkan dimana jika nilai yang diinputkan adalah 1 maka nilai balik adalah nilai itu sendiri. Sedangkan jika yang dimasukkan bukan nilai 1 maka akan dihitung menggunakan rumus faktorial.



## BAB III

### TUGAS PRAKTIKUM

#### 3.1 Tugas 1

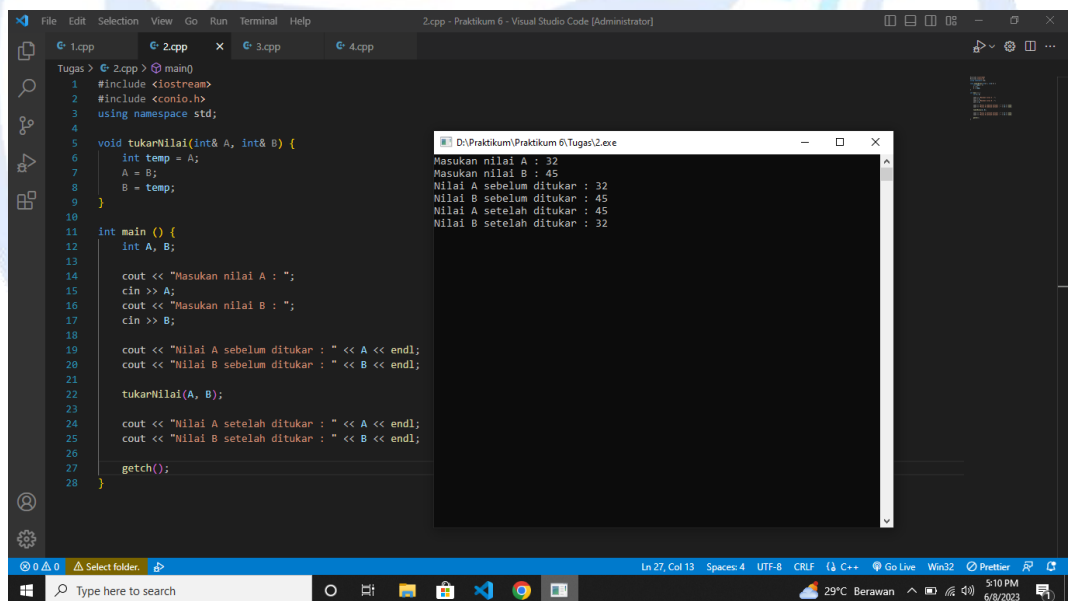


```
1 #include <iostream>
2 #include <conio.h>
3 using namespace std;
4
5 void inputData(float& panjang, float& lebar) {
6     cout << "Masukan Panjang : ";
7     cin >> panjang;
8     cout << "Masukan Lebar : ";
9     cin >> lebar;
10 }
11
12 float hitungLuas(int panjang, int lebar) {
13     float luas = panjang * lebar;
14     return luas;
15 }
16
17 void outputData(float luas) {
18     cout << "Luas Persegi Panjang adalah " << luas << endl;
19 }
20
21 int main () {
22     float panjang, lebar;
23
24     inputData(panjang, lebar);
25
26     float luas = hitungLuas(panjang, lebar);
27
28     outputData(luas);
29     getch();
30 }
```

Output:

```
Masukan Panjang : 50
Masukan Lebar : 20
Luas Persegi Panjang adalah 1000
```

#### 3.2 Tugas 2

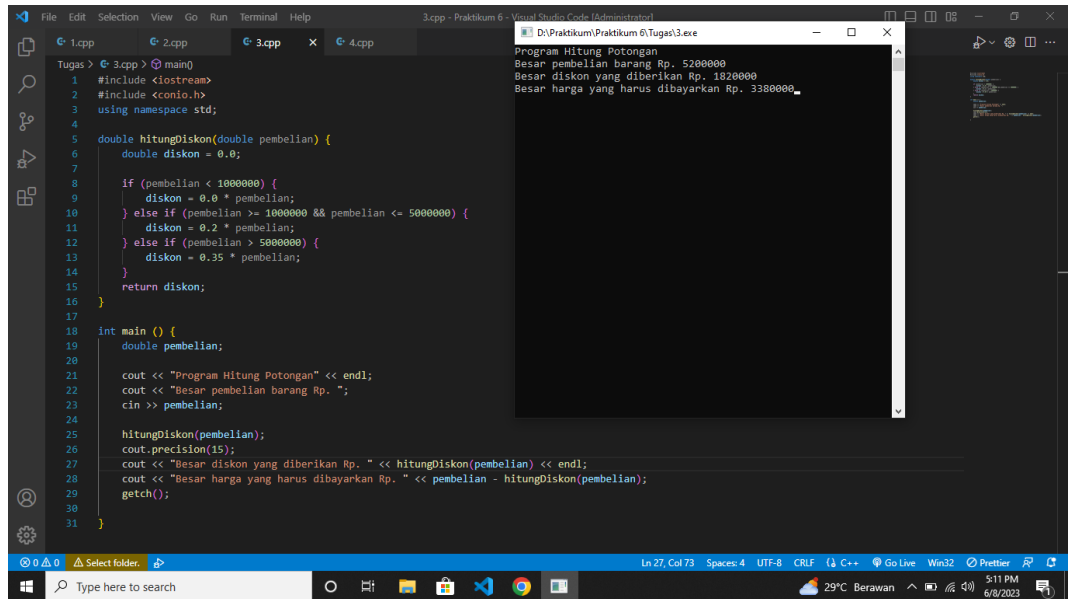


```
1 #include <iostream>
2 #include <conio.h>
3 using namespace std;
4
5 void tukarNilai(int& A, int& B) {
6     int temp = A;
7     A = B;
8     B = temp;
9 }
10
11 int main () {
12     int A, B;
13
14     cout << "Masukan nilai A : ";
15     cin >> A;
16     cout << "Masukan nilai B : ";
17     cin >> B;
18
19     cout << "Nilai A sebelum ditukar : " << A << endl;
20     cout << "Nilai B sebelum ditukar : " << B << endl;
21
22     tukarNilai(A, B);
23
24     cout << "Nilai A setelah ditukar : " << A << endl;
25     cout << "Nilai B setelah ditukar : " << B << endl;
26
27     getch();
28 }
```

Output:

```
Masukan nilai A : 32
Masukan nilai B : 45
Nilai A sebelum ditukar : 32
Nilai B sebelum ditukar : 45
Nilai A setelah ditukar : 45
Nilai B setelah ditukar : 32
```

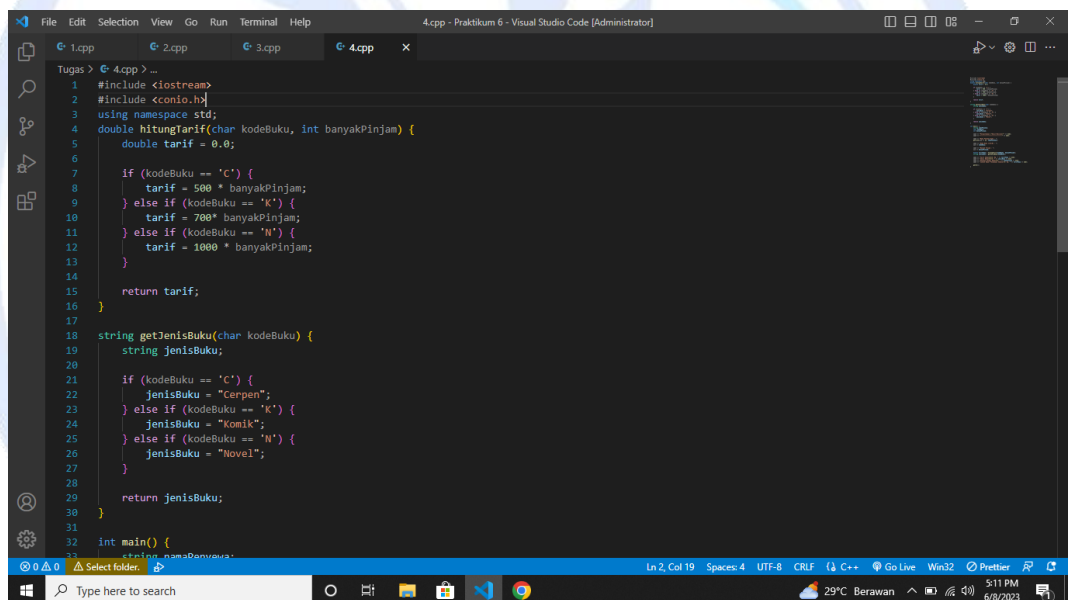
#### 3.3 Tugas3



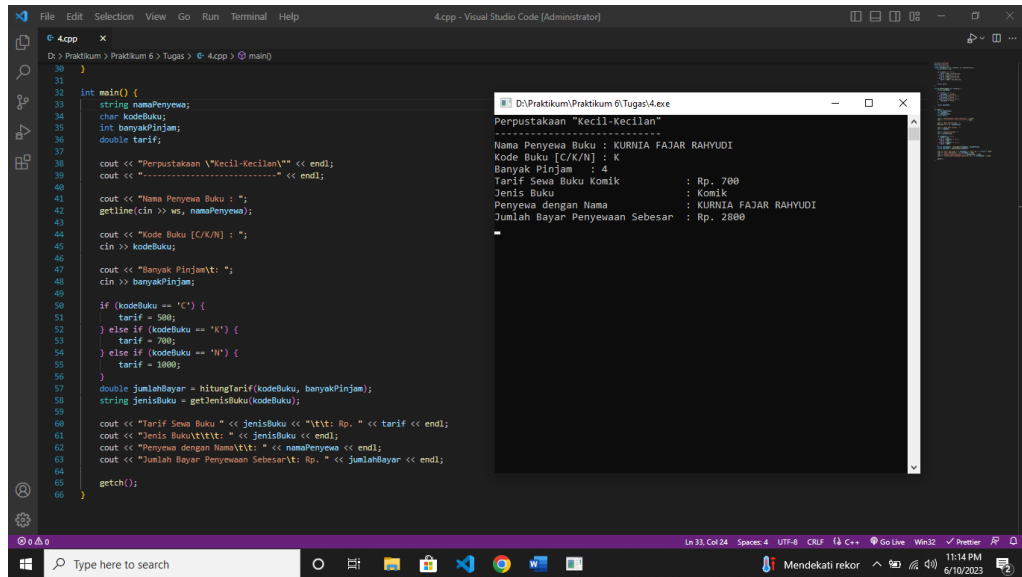
```
File Edit Selection View Go Run Terminal Help
3.cpp - Praktikum 6 - Visual Studio Code [Administrator]
3.cpp x 4.cpp
Tugas > 3.cpp > main()
1 #include <iostream>
2 #include <conio.h>
3 using namespace std;
4
5 double hitungDiskon(double pembelian) {
6     double diskon = 0.0;
7
8     if (pembelian < 1000000) {
9         diskon = 0.0 * pembelian;
10    } else if (pembelian >= 1000000 && pembelian <= 5000000) {
11        diskon = 0.2 * pembelian;
12    } else if (pembelian > 5000000) {
13        diskon = 0.35 * pembelian;
14    }
15    return diskon;
16 }
17
18 int main() {
19     double pembelian;
20
21     cout << "Program Hitung Potongan" << endl;
22     cout << "Besarnya pembelian barang Rp. ";
23     cin >> pembelian;
24
25     hitungDiskon(pembelian);
26     cout.precision(15);
27     cout << "Besarnya diskon yang diberikan Rp. " << hitungDiskon(pembelian) << endl;
28     cout << "Besarnya harga yang harus dibayarkan Rp. " << pembelian - hitungDiskon(pembelian);
29     getch();
30 }
31 }
```

Program Hitung Potongan  
Besarnya pembelian barang Rp. 5200000  
Besarnya diskon yang diberikan Rp. 1040000  
Besarnya harga yang harus dibayarkan Rp. 3380000

### 3.4 Tugas 4



```
File Edit Selection View Go Run Terminal Help
4.cpp - Praktikum 6 - Visual Studio Code [Administrator]
4.cpp x
Tugas > 4.cpp > ...
1 #include <iostream>
2 #include <conio.h>
3 using namespace std;
4 double hitungTarif(char kodeBuku, int banyakPinjam) {
5     double tarif = 0.0;
6
7     if (kodeBuku == 'C') {
8         tarif = 500 * banyakPinjam;
9     } else if (kodeBuku == 'K') {
10        tarif = 700 * banyakPinjam;
11    } else if (kodeBuku == 'N') {
12        tarif = 1000 * banyakPinjam;
13    }
14
15    return tarif;
16 }
17
18 string getJenisBuku(char kodeBuku) {
19     string jenisBuku;
20
21     if (kodeBuku == 'C') {
22         jenisBuku = "Cerpen";
23     } else if (kodeBuku == 'K') {
24         jenisBuku = "Komik";
25     } else if (kodeBuku == 'N') {
26         jenisBuku = "Novel";
27     }
28
29    return jenisBuku;
30 }
31
32 int main() {
33     string namaRanveera;
```



The screenshot shows the Visual Studio Code editor with a C++ file named `4.cpp`. The code is a program for a library system. It prompts the user for the library name, borrower name, book code, number of books borrowed, and book type. It then calculates the total amount to be paid based on the book type and the number of books borrowed.

```
30 }
31
32 int main() {
33     string namaPenyewa;
34     char kodeBuku;
35     int banyakPinjam;
36     double tarif;
37
38     cout << "Perpustakaan \"Kecil-Kecilan\" << endl;
39     cout << "-----" << endl;
40
41     cout << "Nama Penyewa Buku : ";
42     getline(cin >> ws, namaPenyewa);
43
44     cout << "Kode Buku [C/K/N] : ";
45     cin >> kodeBuku;
46
47     cout << "Banyak Pinjam\t: ";
48     cin >> banyakPinjam;
49
50     if (kodeBuku == 'C') {
51         tarif = 500;
52     } else if (kodeBuku == 'K') {
53         tarif = 700;
54     } else if (kodeBuku == 'N') {
55         tarif = 1000;
56     }
57     double jumlahBayar = hitungTarif(kodeBuku, banyakPinjam);
58     string jenisBuku = getJenisBuku(kodeBuku);
59
60     cout << "Tarif Sewa Buku " << jenisBuku << "t\t: Rp. " << tarif << endl;
61     cout << "Jenis Buku\t\t\t: " << jenisBuku << endl;
62     cout << "Penyewa dengan Nama\t\t: " << namaPenyewa << endl;
63     cout << "Jumlah Bayar Penyewaan Sebesar\t: Rp. " << jumlahBayar << endl;
64
65     getch();
66 }
```

The output window shows the following results:

```
Perpustakaan "Kecil-Kecilan"
-----
Nama Penyewa Buku : KURNIA FAJAR RAHYUDI
Kode Buku [C/K/N] : K
Banyak Pinjam : 4
Tarif Sewa Buku Komik : Rp. 700
Jenis Buku : Komik
Penyewa dengan Nama : KURNIA FAJAR RAHYUDI
Jumlah Bayar Penyewaan Sebesar : Rp. 2800
```



## BAB IV

### ANALISA

#### 4.1 Tugas 1

- Poin A

Program ini menggunakan `#include <iostream>` sebagai library untuk menerima input dan memberi output, seperti `cin` dan `cout`. Using `namespace std;` berguna sebagai perintah yang digunakan untuk menyajikan perintah menggunakan seluruh berkas atau fungsi yang menjadi bagian dari namespace `std` yang digunakan.

- Poin B

Fungsi **'inputdata'** dideklarasikan dengan tipe **'void'**, yang berarti fungsi ini tidak mengembalikan nilai. Fungsi ini memiliki dua parameter **'panjang'** dan **'lebar'** yang dinyatakan sebagai referensi (&). Saat fungsi ini dipanggil, nilai panjang dan lebar akan dimasukkan oleh pengguna melalui **'cin'** (input dari keyboard) dan disimpan di dalam variabel yang diacu oleh parameter referensi tersebut.

- Poin C

Fungsi **hitungluas** dideklarasikan dengan tipe **float**, yang berarti fungsi ini mengembalikan nilai bertipe **float**. Fungsi ini memiliki dua parameter **panjang** dan **lebar** yang dinyatakan sebagai variabel lokal. Di dalam fungsi ini, variabel lokal **luas** dideklarasikan dan diinisialisasi dengan hasil perkalian **panjang** dan **lebar**. Nilai **luas** kemudian dikembalikan sebagai hasil fungsi menggunakan pernyataan **return**.

- Poin D

Fungsi **outputdata** dideklarasikan dengan tipe **void**, yang berarti fungsi ini tidak mengembalikan **nilai**. Fungsi ini memiliki satu parameter **luas** yang dinyatakan sebagai variabel lokal. Di dalam fungsi ini, pesan "Luas Persegi Panjang adalah : " dan nilai luas akan ditampilkan ke layar menggunakan **cout**. **endl** digunakan untuk memberikan baris baru setelah menampilkan **luas**.

- Poin E

Fungsi **main** adalah fungsi utama yang akan dieksekusi saat program dijalankan. Di dalam fungsi **main**, variabel lokal **panjang** dan **lebar** dideklarasikan dengan tipe **float** untuk menyimpan input **panjang** dan **lebar** persegi panjang. Fungsi **inputdata** dipanggil untuk meminta pengguna memasukkan **panjang** dan **lebar**. Nilai **panjang** dan **lebar** akan berubah setelah pengguna memasukkan input. Fungsi **hitungluas** dipanggil dengan argumen **panjang** dan **lebar** untuk menghitung **luas** persegi panjang, dan hasilnya disimpan dalam variabel lokal **luas**. Fungsi **outputdata** dipanggil dengan argumen **luas** untuk menampilkan luas persegi panjang ke layar. Akhirnya, nilai 0 dikembalikan oleh fungsi **main** sebagai tanda bahwa program berjalan tanpa masalah.

#### 4.2 Tugas 2

- Poin A

Program ini menggunakan `#include <iostream>` sebagai library untuk menerima input dan memberi output, seperti `cin` dan `cout`, sedangkan `<conio.h>` berisi fungsi-fungsi untuk operasi layar dan keyboard di lingkungan Windows. `using namespace std;` berguna sebagai perintah yang digunakan untuk menyajikan perintah menggunakan seluruh berkas atau fungsi yang menjadi bagian dari namespace `std` yang digunakan.

- Poin B

Fungsi **tukarletak** dideklarasikan dengan tipe **void**, yang berarti fungsi ini tidak mengembalikan nilai. Fungsi ini memiliki dua parameter **a** dan **b** yang dinyatakan sebagai referensi (&). Di dalam fungsi ini, nilai **a** disimpan dalam variabel **temp**. Kemudian, nilai **a** diperbarui dengan nilai **b**, dan nilai **b** diperbarui dengan nilai **temp**. Dengan demikian, fungsi ini bertujuan untuk menukar nilai antara dua variabel yang diberikan.

- Poin C

Fungsi **main** adalah fungsi utama yang akan dieksekusi saat program dijalankan. Di dalam fungsi **main**, dua variabel lokal **bilanganA** dan **bilanganB** dideklarasikan untuk menyimpan input dari pengguna. Pesan "Masukkan Bilangan A : " ditampilkan menggunakan **cout**, dan nilai

bilanganA diinputkan oleh pengguna melalui **cin**. Pesan yang serupa juga ditampilkan untuk menerima input bilanganB.

Setelah itu, pesan "Bilangan Sebelum ditukar A = " diikuti dengan nilai bilanganA, dan pesan "Bilangan Setelah ditukar A = " diikuti dengan nilai bilanganB ditampilkan menggunakan **cout**. Kemudian, fungsi **tukarletak** dipanggil dengan argumen **bilanganA** dan **bilanganB** untuk menukar nilai antara dua variabel tersebut.

Selanjutnya, pesan "Bilangan Sebelum ditukar B = " diikuti dengan nilai bilanganA, dan pesan "Bilangan Setelah ditukar B = " diikuti dengan nilai bilanganB ditampilkan menggunakan **cout**. Ini menunjukkan hasil pertukaran nilai setelah pemanggilan fungsi **tukarletak**.

**getch()** digunakan untuk menahan tampilan layar sehingga pengguna dapat melihat hasilnya sebelum program selesai dieksekusi. Fungsi **getch()** diambil dari header **<conio.h>**.

#### 4.3 Tugas 3

- Poin A

Program ini menggunakan **#include <iostream>** sebagai library untuk menerima input dan memberi output, seperti **cin** dan **cout**. Using namespace **std**; berguna sebagai perintah yang digunakan untuk menyajikan perintah menggunakan seluruh berkas atau fungsi yang menjadi bagian dari namespace **std** yang digunakan.

- Poin B

Fungsi **hitungdiskon** dideklarasikan dengan tipe **double**, yang berarti fungsi ini mengembalikan nilai bertipe **double**. Fungsi ini memiliki satu parameter **pembelian** yang dinyatakan sebagai variabel lokal. Di dalam fungsi ini, variabel lokal **diskon** dideklarasikan dan diinisialisasi dengan nilai 0.0.

Kemudian, dilakukan pengecekan dengan menggunakan struktur pengkondisian **if** dan **else if**. Jika pembelian berada di antara 1.000.000 dan 5.000.000, maka **diskon** akan dihitung sebagai 20% dari **pembelian**. Jika **pembelian** lebih besar atau sama dengan 5.000.000, maka **diskon**



akan dihitung sebagai 35% dari **pembelian**. Jika kondisi-kondisi tersebut tidak terpenuhi, maka nilai **diskon** tetap 0.

Terakhir, nilai **diskon** dikembalikan sebagai hasil fungsi menggunakan pernyataan **return**.

- Poin C

Fungsi **main** adalah fungsi utama yang akan dieksekusi saat program dijalankan. Di dalam fungsi **main**, variabel lokal **pembelian** dideklarasikan dengan tipe **double** untuk menyimpan input jumlah pembelian.

Pesan "Program Hitung Potongan" ditampilkan ke layar menggunakan **cout**. Kemudian, pesan "Besar Pembelian yang diberikan Rp. " ditampilkan menggunakan **cout**, dan nilai pembelian dimasukkan oleh pengguna melalui **cin**.

Selanjutnya, fungsi **hitungdiskon** dipanggil dengan argumen **pembelian** untuk menghitung **diskon** berdasarkan jumlah pembelian. Hasil **diskon** disimpan dalam variabel lokal **diskon**. Jumlah yang harus dibayarkan (hargabayar) dihitung dengan mengurangi **diskon** dari **pembelian**.

Nilai presisi output untuk angka desimal ditetapkan menggunakan **cout.precision(15)**, yang akan menampilkan 15 digit angka desimal setelah koma. Pesan "Besar Diskon yang diberikan Rp. " diikuti dengan nilai **diskon** dan pesan "Besar Harga yang harus dibayarkan Rp. " diikuti dengan nilai **hargabayar** ditampilkan ke layar menggunakan **cout**.

Terakhir, nilai 0 dikembalikan oleh fungsi **main** sebagai tanda bahwa program berjalan tanpa masalah.



#### 4.4 Tugas 4

- Poin A

Program ini menggunakan `#include <iostream>` sebagai library untuk menerima input dan memberi output, seperti `cin` dan `cout`. Using namespace `std`; berguna sebagai perintah yang digunakan untuk menyajikan perintah menggunakan seluruh berkas atau fungsi yang menjadi bagian dari namespace `std` yang digunakan.

- Poin B

Fungsi **nameSewa** dideklarasikan dengan tipe **void**, yang berarti fungsi ini tidak mengembalikan nilai. Fungsi ini memiliki satu parameter **nama** yang dinyatakan sebagai referensi (&). Di dalam fungsi ini, pesan "Nama Penyewa Buku: " ditampilkan menggunakan **cout**, dan input dari pengguna diambil menggunakan **getline** untuk menghindari masalah dengan karakter tersisa di buffer setelah menggunakan **cin**.

- Poin C

Fungsi **tarif** dideklarasikan dengan tipe **int**, yang berarti fungsi ini mengembalikan nilai bertipe **int**. Fungsi ini memiliki satu parameter **kodeBuku** yang dinyatakan sebagai variabel lokal. Di dalam fungsi ini, menggunakan struktur pengkondisian **if** dan **else if** untuk menentukan tarif sewa berdasarkan **kodeBuku**. Jika **kodeBuku** adalah 'C', tarifnya adalah 500. Jika **kodeBuku** adalah 'K', tarifnya adalah 700. Jika **kodeBuku** adalah 'N', tarifnya adalah 1000. Jika kondisi-kondisi tersebut tidak terpenuhi, maka tarifnya adalah 0.

- Poin D

Fungsi **main** adalah fungsi utama yang akan dieksekusi saat program dijalankan. Di dalam fungsi **main**, variabel lokal **nama** dengan tipe **string**, **kodeBuku** dengan tipe **char**, dan **banyakPinjam** dengan tipe **int** dideklarasikan.

Pesan "Perpustakaan "Kecil-Kecilan"" dan "-----" ditampilkan ke layar menggunakan **cout**.

Fungsi **nameSewa** dipanggil untuk meminta dan menyimpan nama penyewa buku.

Pesan "Kode Buku [C/K/N]: " ditampilkan menggunakan **cout**, dan nilai **kodeBuku** dimasukkan oleh pengguna melalui **cin**.

Pesan "Banyak Pinjam: " ditampilkan menggunakan **cout**, dan nilai **banyakPinjam** dimasukkan oleh pengguna melalui **cin**.

Nilai **tarifSewa** dihitung dengan memanggil fungsi **tarif** dengan argumen **toupper(kodeBuku)**, yang mengubah karakter **kodeBuku** menjadi huruf besar untuk menangani masukan huruf kecil.

Variabel **jenisBuku** dengan tipe **string** dideklarasikan untuk menyimpan jenis buku berdasarkan **kodeBuku**.

Jika **tarifSewa** lebih dari 0, berarti **kodeBuku** valid, maka dilakukan pengecekan menggunakan struktur pengkondisian **if** dan **else if**.

Berdasarkan nilai **kodeBuku**, variabel **jenisBuku** diisi dengan jenis buku yang sesuai.

Jumlah **pembayaran** (**jumlahBayar**) dihitung dengan mengalikan **tarifSewa** dengan **banyakPinjam**.

Pesan "Tarif Sewa Rp. " diikuti dengan nilai **tarifSewa**, pesan "Jenis Buku: " diikuti dengan nilai **jenisBuku**, pesan "Penyewa dengan Nama " diikuti dengan nilai nama, dan pesan "Jumlah Bayar Penyewaan Sebesar Rp. " diikuti dengan nilai **jumlahBayar** ditampilkan ke layar menggunakan **cout**.

Jika **tarifSewa** adalah 0, berarti **kodeBuku** tidak valid, maka pesan "Kode Buku tidak valid." ditampilkan ke layar menggunakan **cout**.

Terakhir, nilai 0 dikembalikan oleh fungsi **main** sebagai tanda bahwa program berjalan tanpa masalah.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Prosedur/Fungsi adalah suatu program terpisah dalam blok sendiri yang berfungsi sebagai subprogram (program bagian). Modul pada bahasa C++ dikenal dengan nama fungsi (function). Kegunaan dari fungsi ini adalah untuk mengurangi pengulangan penulisan program yang berulang atau sama dan program menjadi lebih terstruktur, sehingga mudah dipahami dan dapat lebih dikembangkan.

Perbedaan antara fungsi dan prosedur yaitu Prosedur dalam Bahasa C++ diawali dengan tipe data “void” yang artinya kosong. Sedangkan fungsi dalam Bahasa C++ diawali dengan sebuah tipe data seperti int, Boolean, float, dan fungsi memiliki sebuah nilai kembalian yang di tandai dengan “return”.

Rekursif adalah memanggil dirinya sendiri, artinya fungsi tersebut di panggil pada bagian tubuh fungsi atau isi fungsi. Hampir sama seperti perulangan, fungsi rekursif juga terdapat batasan kapan rekursif tersebut akan berhenti. Jika tidak maka rekursif tersebut tidak akan berhenti(infinite loop)

#### **5.2 Saran**

Dalam membuat program C++ perlu diperhatikan huruf kecil atau besar nama array, indeks yang diinput juga harus sesuai dengan yang telah dibuat karena jika indeksnya lebih dari yang ada akan berpengaruh pada programnya sendiri serta bisa sampai menyebabkan error. Selain itu juga harus lebih sering berlatih dan belajar membuat program menggunakan fungsi dan prosedur agar lebih cepat dalam memahami bagaimana pengaplikasian fungsi dan prosedur dalam C++.

## DAFTAR PUSTAKA

Anonim. (2003). *MODUL 5 PRAKTIKUM ALGORITMA DAN PEMECAHAN MASALAH*. Laboratorium Komputer dan Jaringan.

Kelasprogrammer. (2020, Januari 23). Retrieved from <https://kelasprogrammer.com/contoh-program-cpp-fungsi-dan-prosedur/>

Pintarkom. (2019, November 18). Retrieved from <https://pintarkom.com/fungsi-pada-c-plus/>

