

JURNAL PRAKTIKUM
PRAKTIKUM ALGORITMA DAN PEMECAHAN MASALAH
MODUL 1 PENGENALAN BAHASA C++



Nama : Kurnia Fajar Rahyudi Putra

No. BP : 2211512013

Hari/Tanggal : Senin/8 Mei 2021

Shift : 1

Dosen : Dodon Yendri, M.Kom

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

BAB I PENDAHULUAN

1.1 Tujuan

Tujuan dari praktikum ini adalah :

1. Pratikan mengenali batang tubuh program bahasa C++.
2. Pratikan mengetahui aturan pemberian nama.
3. Pratikan dapat mempraktekkan variabel, konstanta, dan operator.
4. Pratikan dapat membuat file kerja dengan Bahasa C++.
5. Pratikan dapat membuat program baru, meng-compile, menyimpan dan menjalankan program yang telah dibuat.
6. Pratikan dapat memanggil/membuka kembali program yang telah disimpan.
7. Pratikan dapat membuat sebuah program menjadi program yang execute.

1.2 Landasan Teori

1.2.1 Program Komputer

Program Komputer adalah suatu perangkat lunak (*software*) yang digunakan untuk keperluan-keperluan aplikatif tertentu di berbagai bidang, baik di lingkungan perusahaan, pendidikan ataupun yang lainnya.

Perangkat lunak itu sendiri merupakan suatu susunan kode-kode program yang ditulis oleh salah satu bahasa pemrograman tertentu dan telah dikompilasi melalui kompilator yang sesuai dengan bahasa pemrograman tersebut.

1.2.2 Bahasa Pemrograman

Bahasa pemrograman adalah suatu kumpulan kata (perintah) yang siap digunakan untuk menulis suatu kode program sehingga kode-kode program yang kita tulis tersebut akan dapat dikenali oleh kompilator yang sesuai.

1.2.3 Kompilator

Kompilator (*compiler*) dapat diartikan sebagai suatu penerjemah, yaitu kumpulan kode program yang ditulis dalam suatu bahasa pemrograman tertentu akan diterjemahkan oleh kompilator ke dalam bahasa *assembly*; yang selanjutnya akan diterjemahkan lagi menjadi kode objek sehingga perintah-perintahnya akan dikenali oleh komputer (dalam hal ini mesin).

1.2.4 Algoritma Komputer

Algoritma merupakan skema kerja komputer yang dilengkapi dengan perangkat otak (brainware), perangkat lunak (software) dan perangkat keras (hardware). Sistem komputer tidak akan bertindak dengan baik apabila tanpa ketiga perangkat tersebut.

Algoritma juga disebutkan sebagai suatu proses serangkaian perintah-perintah yang berkonsolidasi, atau cara spesial dalam memecahkan persoalan yang praktis (webster kamus).

1.2.5 Pengantar C++

C++ diciptakan oleh Bjarne Stroustrup di laboratorium Bell pada awal tahun 1980-an, sebagai pengembangan dari bahasa C dan Simula. Saat ini, C++ merupakan salah satu bahasa yang paling populer untuk pengembangan software berbasis OOP. Kompiler untuk C++ telah banyak beredar di pasaran. Software developer yang paling diminati adalah Borland Inc. dan Microsoft Corp. Produk dari Borland untuk kompiler C++ adalah Turbo C++, Borland C++, Borland C++ Builder. Sedangkan dari Microsoft adalah Ms. Visual C++. Walaupun banyak kompiler yang tersedia, namun pada intinya bahasa pemrograman yang dipakai adalah C++. Sebelum mulai melakukan kode program, sebaiknya diingat bahwa C++ bersifat “case sensitive”, yang artinya huruf besar dan huruf kecil dibedakan.

1.2.6 Batang Tubuh bahasa C++

Setiap program C++ mempunyai bentuk umum seperti berikut, yakni :

```
# preprocessor directive(include)
int main()
{
```

```
//batang tubuh program utama
}
```

Penjelasan :

▪ include

Merupakan salah satu pengarah *preprocessor directive* yang tersedia di C++. *Preprocessor* selalu dijalankan terlebih dahulu pada saat proses kompilasi terjadi.

Bentuk umum *include* adalah :

#include <nama_file> atau

`#include` “nama_file” `include` tidak diakhiri dengan semicolon (;), karena bentuk `include` bukan merupakan sebuah pernyataan melainkan *preprocessor directive*. Baris tersebut menginstruksikan pada compiler yang menyisipkan file lain, dalam hal ini file yang berakhiran .h (file heder) yaitu file yang berisi sebuah deklarasi, contohnya :

- `#include <iostream.h>` : diperlukan pada program yang melibatkan objek `cout` (C++)
- `#include <stdio.h>` : diperlukan pada program yang melibatkan objek `cout` (C)
- `#include “conio.h”` : diperlukan bila menggunakan `clrscr()`, yaitu perintah membersihkan layar

▪ `main()`

Fungsi ini merupakan awal dan akhir program C++, **main** adalah nama judul fungsi. dimana batang tubuh program utama berada pada fungsi tersebut. ▪ semicolon (;) Tanda semicolon”;" berfungsi untuk mengakhiri sebuah pernyataan, dimana setiap pernyataan harus diakhiri dengan semicolon.

Contoh program bahasa C++ :

```
/* contoh penulisan program C++ yang baik */
#include <iostream>
#include <conio> int
main()
{
cout << “selamat datang di bahasa C++ \n”;
}
```

1.2.8 Tipe Data dalam Bahasa Pemrograman C++

Tipe data merupakan bagian program yang paling penting karena tipe data akan mempengaruhi setiap instruksi yang akan dilaksanakan oleh komputer. Pemilihan tipe data yang tepat akan membuat proses operasi data menjadi lebih efisien dan efektif.

No	Tipe Data	Ukuran	Jangkauan	Keterangan
1	Char	1 byte	- 128 .. 127	Karakter/string
2	Int	2 byte	- 32768 .. 32767	Integer/bilangan bulat

3	float	4 byte	- 3.4E-38 .. 3.4E+38	Float/bilangan pecahan
4	double	8 byte	- 1.7E-308 ..d 1.7+308	Pecahan presisi ganda
5	Void	0 byte	-	Tidak bertipe

1.2.9 Variabel dalam Bahasa Pemrograman C++

Variabel adalah suatu pengenal (*identifier*) yang digunakan untuk mewakili suatu nilai tertentu di dalam proses program. Berbeda dengan konstanta yang nilainya selalu tetap, nilai dari suatu variabel bisa diubah-ubah sesuai kebutuhan. Nama dari suatu variabel dapat ditentukan sendiri oleh pemrogram dengan aturan sebagai berikut :

- Terdiri dari gabungan huruf dan angka dengan karakter pertama harus berupa huruf. Bahasa C++ bersifat *case-sensitive* artinya huruf besar dan kecil dianggap berbeda. Jadi antara **nama**, **NAMA** dan **Nama** dianggap berbeda, sedangkan pada bahasa Pascal tidak.
- Tidak boleh mengandung spasi.
- Tidak boleh mengandung simbol-simbol khusus, kecuali garis bawah (*underscore*). Yang termasuk simbol khusus dan tidak diperbolehkan antara lain : \$, ?, %, #, !, &, *, (,), -, +, = dan lain sebagainya.
- Panjangnya bebas, tetapi hanya 32 karakter pertama yang terpakai.

Contoh penamaan variabel **yang benar** : NIM, a, x, nama_mhs,

Contoh penamaan variabel **yang salah** : %nilai_mahasiswa, 80mahasiswa, ratarata (ada spasi penting diketahui !).

1.2.10 Deklarasi dalam Bahasa Pemrograman C++

Deklarasi diperlukan bila kita akan menggunakan pengenal (*identifier*) dalam program. Identifier dapat berupa variabel, konstanta dan fungsi.

▪ Deklarasi Variabel

Deklarasi variabel dalam bahasa C++ :

nama_tipe nama_variabel; Contoh : `int y; // deklarasi variabel y bertipe`

`integer char y, huruf, bp[10]; // deklarasi variabel y dan huruf`

`bertipe char float nilai; // deklarasi variabel nilai bertipe float`

`double venus; // deklarasi variabel venus bertipe double`


```
int array[5][4]; // deklarasi array bertipe integer
char *y; // deklarasi pointer y bertipe char
```

- Deklarasi Konstanta

Dalam bahasa C++ konstanta dideklarasikan menggunakan *preprocessor* `#define`.

Contoh:

```
#define PHI 3.14
#define bp "0810452099"
#define nama "Megawati"
```

- Deklarasi Fungsi

Fungsi merupakan bagian yang terpisah dari program dan dapat diaktifkan atau dipanggil di manapun di dalam program. Umumnya dalam bahasa pemrograman, beberapa fungsi yang sudah disediakan bahasa tersebut. Sebagai contoh dalam bahasa C++, fungsi pustaka `cout`, `cin` atau `printf()`, `scanf()` (untuk bahasa C), `getch()` dan lain sebagainya. Dalam bahasa Pascal seperti `write()` atau `writeln()`, `read` atau `readln()`. Untuk menggunakan fungsi tersebut programmer dapat langsung menuliskan fungsi tersebut.

Disamping fungsi yang sudah disediakan oleh bahasa pemrograman, programmer dapat pula membuat fungsi sendiri sesuai dengan kebutuhannya. Untuk mendefinisikan sendiri fungsi tersebut, harus dideklarasikan terlebih dahulu dalam program yang dibuat kemudian baru bisa menggunakannya. Berikut adalah cara mendeklarasikan fungsi yang dibuat sendiri oleh programmer.

Deklarasi fungsi dalam bahasa C++ :

tipe_fungsi nama_fungsi(parameter_fungsi); Contoh

:

```
float uas(int nilai);
void output(); int
kali(int x, int y);
```

1.2.12 Operator dalam Bahasa Pemrograman C++

- Operator penugasan (*assignment operator*)

Operator penugasan dalam bahasa C++ berupa tanda sama dengan ("`=`").

Contoh : harga

= 1500; kali =

x * y;

Artinya : variable “harga” diisi dengan 1500 dan variable “kali” diisi dengan hasil perkalian antara x dan y.

Sedangkan dalam bahasa Pascal, operator penugasan dengan menggunakan tanda titik dua sama dengan (“:=”). Contoh

:

Harga := 1500;

Kali := x * y;

▪ Operator aritmatika atau matematika

Tujuan	Operator	Contoh
Penjumlahan	+	luas = pj + lb;
Pengurangan	-	g_bersih = gaji – pot
Perkalian	*	bobot = sks * mutu
Pembagian	/	Na = (uts + uas)/2
Sisa bagi	%	if (bil % 2) = 0

▪ Operator hubungan (*relational operator*)

Operator hubungan berfungsi untuk membandingkan antara dua buah *operand* atau lebih

Tujuan	Operator	Contoh
Kurang dari	<	a < b
Kurang dari sama dengan	<=	5 <=10
Lebih dari	>	15 > 10
Lebih dari sama dengan	>=	ip >= 3.00
Sama dengan	=	a = 100
Tidak sama dengan	!=	‘A’ != ‘a’

▪ Operator logika (*logical operator*)

Operator logika berfungsi untuk membandingkan logika hasil dari operator-operator hubungan. Operator logika terdiri dari 3 macam yaitu :

Tujuan	Operator	Contoh
Dan	&&	a && b
Atau		'a' 'b'
Bukan	!	!(eof)

▪ Operator Bitwise

Operator ini digunakan untuk memanipulasi bit-bit nilai yang ada pada memori. Operator bitwise hanya ada pada ++bahasa C. Ada enam macam operator bitwise, yaitu :

>> : Pergeseran ke bit kanan

<< : Pergeseran ke bit kiri

& : Bitwise AND

^ : Bitwise XOR

| : Bitwise OR

~ : Bitwise NOT

▪ Operator Unary

Operator ini merupakan operator yang hanya membutuhkan satu operand saja. Operator Unary juga hanya ada pada bahasa C. Terdapat beberapa operator unary yaitu :

Operator	Keterangan	Posisi	Contoh
-	Unary minus	Sebelum	X+ -Y
++	Peningkatan dengan penambahan nilai 1	Sebelum dan sesudah	Y++
--	Penurunan dengan pengurangan nilai 1	Sebelum dan sesudah	Y--
Sizeof	Ukuran dari operand dalam byte	Sebelum	sizeof(i)
!	Unary NOT	Sebelum	!Y
~	Bitwise NOT	Sebelum	~Y
&	Menghasilkan alamat memori operand	sebelum	&Y

*	Menghasilkan nilai dari pointer	sebelum	*Y
---	---------------------------------	---------	----

1.2.11 Input/Output dalam bahasa Pemrograman C++

▪ Input

Pada C++ terdapat beberapa instruksi dalam memasukan data, dimana beberapa fungsi yang dapat kita gunakan adalah :

a. cin;

- Fungsi pustaka cin digunakan untuk menginput data berupa data numerik, karakter dan string. (Sedangkan pada C digunakan fungsi scan() memakai penentu format contoh : %i (integer), %c (char), dan fungsi scanf() tidak memerlukan penentu lebar field. Scanf() akan memberi pergantian baris secara otomatis).

Pemakaian fungsi scanf(), variabelnya harus menggunakan operator alamat &. b.

gets()

- Fungsi gets() digunakan untuk memasukkan data bertipe karakter dan tidak dapat digunakan untuk memasukkan data numerik.
- Harus diakhiri dengan penekanan tombol enter
- Kursor secara otomatis akan pindah baris
- Tidak memerlukan penentu format

c. getchar()

- Fungsi getchar() digunakan untuk membaca data yang bertipe karakter
- Harus diakhiri dengan penekanan tombol enter
- Karakter yang dimasukkan terlihat pada layar
- Pergantian baris secara otomatis

d. getch() dan getche()

- Fungsi getch() dan getche() digunakan untuk membaca data karakter.
- Karakter yang dimasukkan tidak perlu diakhiri dengan penekanan tombol enter.
- Tidak memberikan efek pergantian baris secara otomatis
- Jika menggunakan fungsi getch() karakter yang dimasukkan tidak akan ditampilkan pada layer sehingga sering digunakan untuk meminta inputan berupa password.
- Sedangkan pada getche() karakter yang dimasukkan akan ditampilkan pada layar.

▪ Output

Dalam C++ instruksi untuk menampilkan data yaitu : cout.

Contoh : cout<<nama;

cout<<"Bahasa C++";

Sedangkan pada C untuk menampilkan data ada beberapa cara, yakni :

- a. Fungsi printf() digunakan untuk menampilkan semua jenis data (numeric dan karakter)
- b. Fungsi puts() digunakan untuk menampilkan data string dan secara otomatis akan diakhiri dengan perpindahan baris
- c. Fungsi putchar() digunakan untuk menampilkan sebuah karakter. Dalam menampilkan bilangan pecahan, terdapat cara dalam mengatur tampilan digitnya, dimana bentuk umum pengaturan bilangan pecahan tersebut adalah :

Bentuk Umum :

printf(“%x.yf”, argument); dimana

:

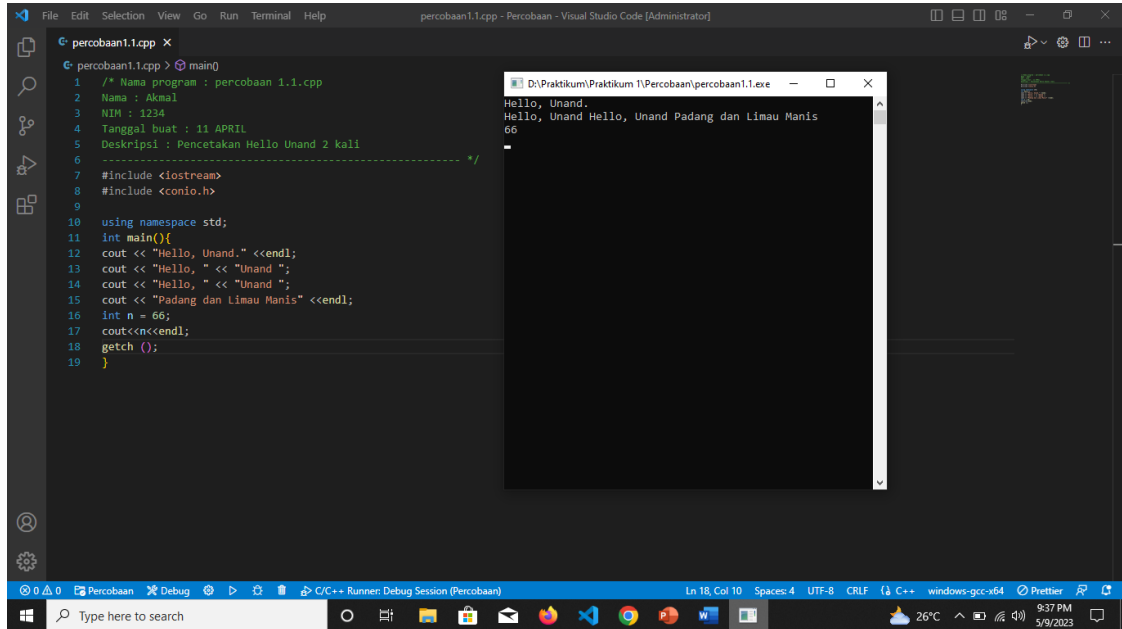
x : menyatakan panjang range y : menyatakan jumlah digit di belakang koma. Argument : nilai atau variable yang akan ditampilkan.

Contoh : printf(“%4.2f”, nilai);

Artinya variable nilai akan ditampilkan sebanyak 4 digit dengan 2 digit di belakang koma.

BAB II PERCOBAAN

2.1 Percobaan 1.1



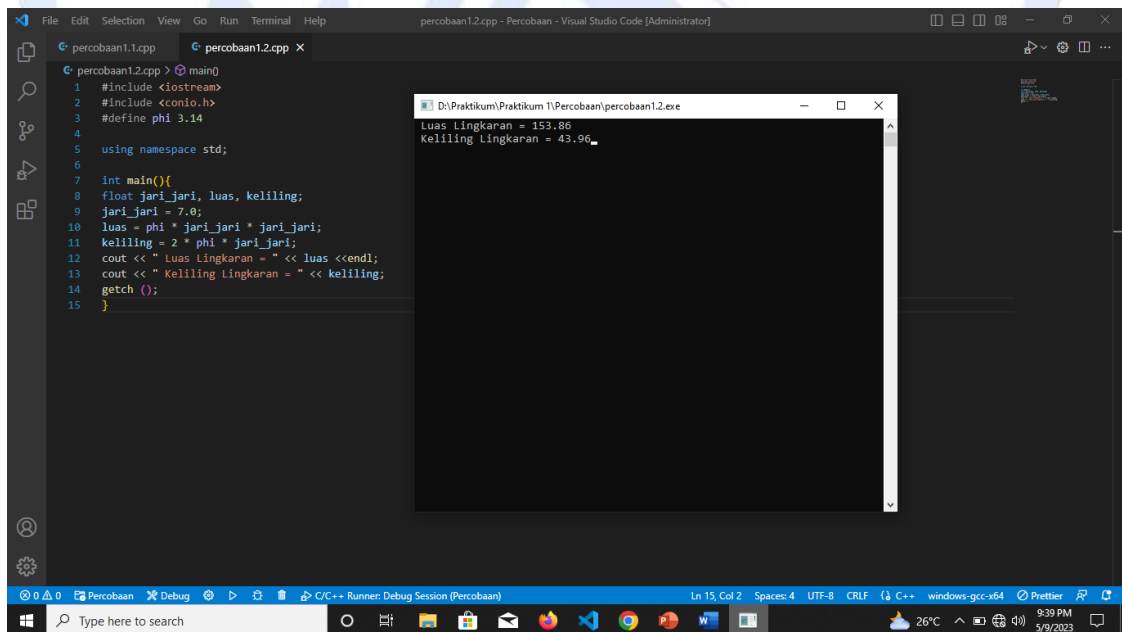
```
1  /* Nama program : percobaan 1.1.cpp
2  Nama : Akmal
3  NIM : 1234
4  Tanggal buat : 11 APRIL
5  Deskripsi : Pencetakan Hello Unand 2 kali
6  ----- */
7  #include <iostream>
8  #include <conio.h>
9
10 using namespace std;
11 int main(){
12     cout << "Hello, Unand." << endl;
13     cout << "Hello, " << "Unand ";
14     cout << "Hello, " << "Unand ";
15     cout << "Padang dan Limau Manis" << endl;
16     int n = 66;
17     cout << n << endl;
18     getch ();
19 }
```

D:\Praktikum\Praktikum 1\percobaan\percobaan1.1.exe

Hello, Unand.
Hello, Unand Hello, Unand Padang dan Limau Manis
66

Pencetakan Hello Unand 2 kali

2.2 Percobaan 1.2



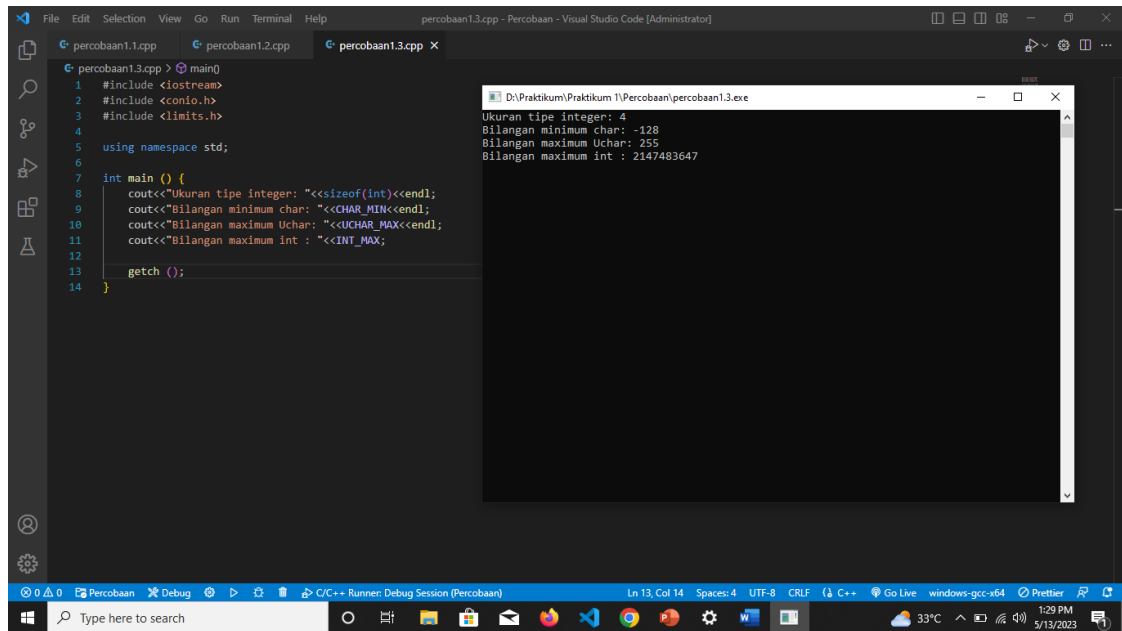
```
1  #include <iostream>
2  #include <conio.h>
3  #define phi 3.14
4
5  using namespace std;
6
7  int main(){
8     float jari_jari, luas, keliling;
9     jari_jari = 7.0;
10     luas = phi * jari_jari * jari_jari;
11     keliling = 2 * phi * jari_jari;
12     cout << "Luas Lingkaran = " << luas << endl;
13     cout << "Keliling Lingkaran = " << keliling;
14     getch ();
15 }
```

D:\Praktikum\Praktikum 1\percobaan\percobaan1.2.exe

Luas Lingkaran = 153.86
Keliling Lingkaran = 43.96

Mencari Luas Lingkaran dan Keliling Lingkaran

2.3 Percobaan 1.3



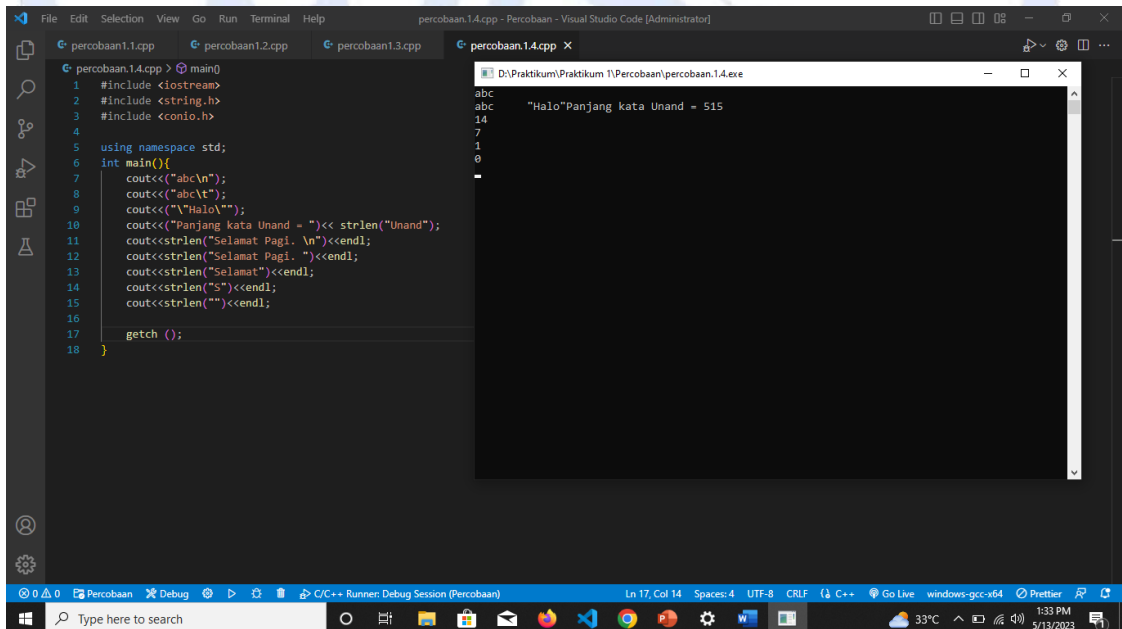
```
1 #include <iostream>
2 #include <conio.h>
3 #include <limits.h>
4
5 using namespace std;
6
7 int main () {
8     cout<<"Ukuran tipe integer: "<<sizeof(int)<<endl;
9     cout<<"Bilangan minimum char: "<<CHAR_MIN<<endl;
10    cout<<"Bilangan maximum Uchar: "<<UCHAR_MAX<<endl;
11    cout<<"Bilangan maximum int : "<<INT_MAX;
12
13    getch ();
14 }
```

Output:

```
Ukuran tipe integer: 4
Bilangan minimum char: -128
Bilangan maximum Uchar: 255
Bilangan maximum int : 2147483647
```

Mencari nilai maksimum atau minimum

2.4 Percobaan 1.4



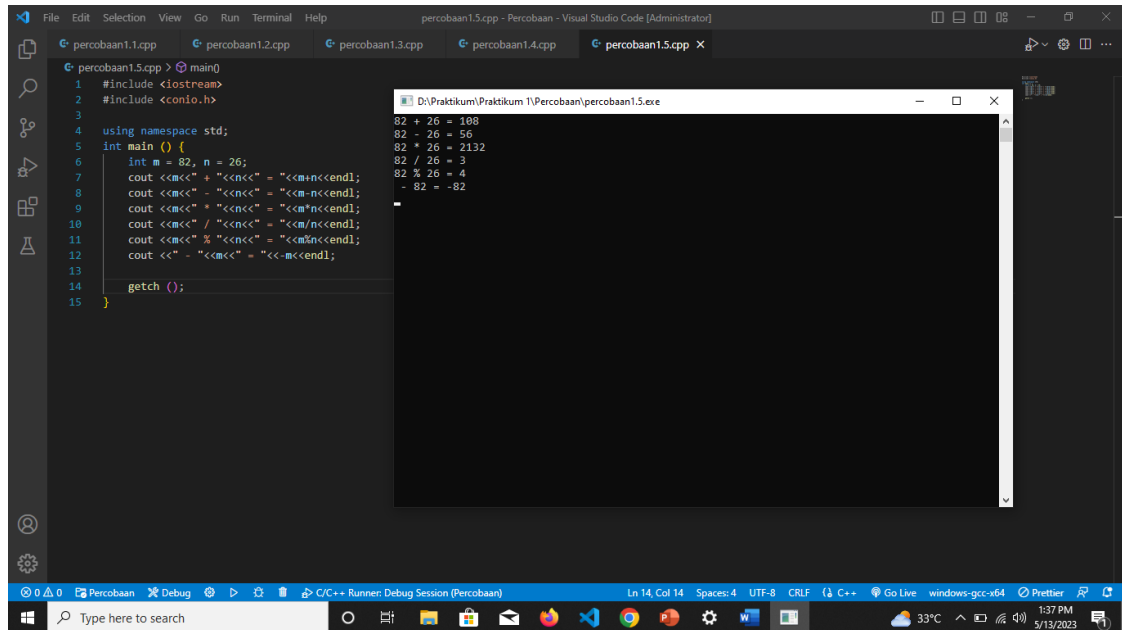
```
1 #include <iostream>
2 #include <string.h>
3 #include <conio.h>
4
5 using namespace std;
6
7 int main(){
8     cout<<("abc\n");
9     cout<<("abc\t");
10    cout<<("\nHalo");
11    cout<<("Panjang kata Unand = "<< strlen("Unand");
12    cout<<strlen("Selamat Pagi. \n")<<endl;
13    cout<<strlen("Selamat Pagi. ")<<endl;
14    cout<<strlen("Selamat")<<endl;
15    cout<<strlen("s")<<endl;
16    cout<<strlen("")<<endl;
17
18    getch ();
19 }
```

Output:

```
abc
abc      "Halo"Panjang kata Unand = 515
14
7
1
0
-
```

Mencari panjang kata atau banyak kata

2.5 Percobaan 2.5



The screenshot shows a Visual Studio Code editor with a C++ file named `percobaan1.5.cpp`. The code defines a `main` function that performs various arithmetic operations on the variables `m = 82` and `n = 26`. The output window displays the results of these operations.

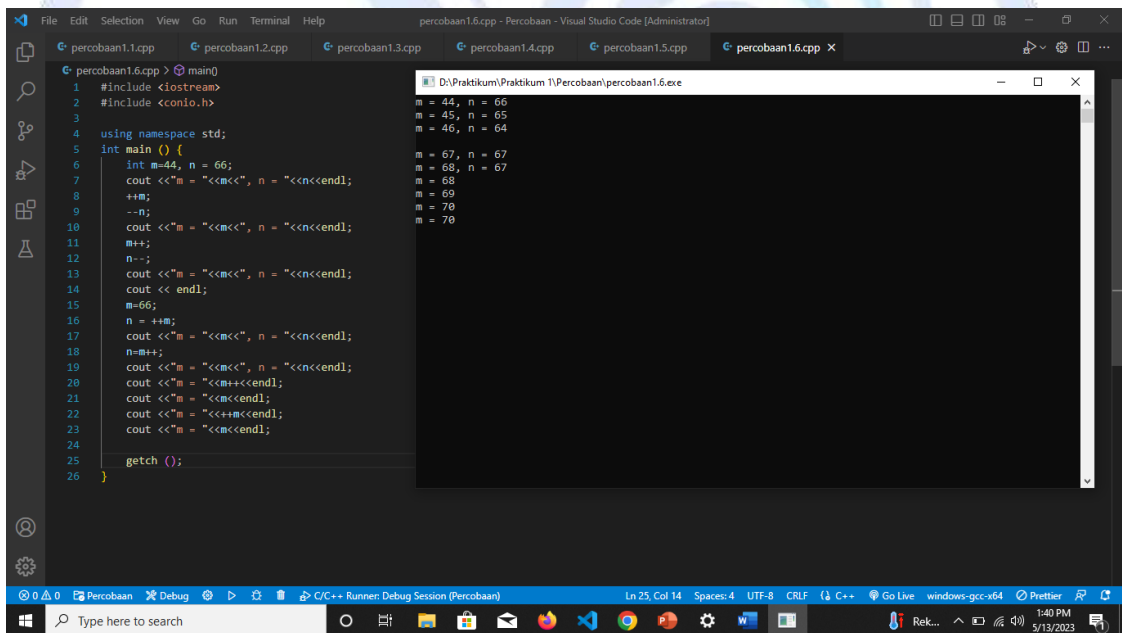
```
1 #include <iostream>
2 #include <conio.h>
3
4 using namespace std;
5 int main () {
6     int m = 82, n = 26;
7     cout << m << " + " << n << " = " << m+n << endl;
8     cout << m << " - " << n << " = " << m-n << endl;
9     cout << m << " * " << n << " = " << m*n << endl;
10    cout << m << " / " << n << " = " << m/n << endl;
11    cout << m << " % " << n << " = " << m%n << endl;
12    cout << " - " << m << " = " << -m << endl;
13
14    getch ();
15 }
```

Output:

```
82 + 26 = 108
82 - 26 = 56
82 * 26 = 2132
82 / 26 = 3
82 % 26 = 4
- 82 = -82
```

Menghitung Aritmatika

2.6 Percobaan 2.6



The screenshot shows a Visual Studio Code editor with a C++ file named `percobaan1.6.cpp`. The code demonstrates the use of increment (`++`) and decrement (`--`) operators on variables `m` and `n`. The output window shows the values of `m` and `n` at each step of the operations.

```
1 #include <iostream>
2 #include <conio.h>
3
4 using namespace std;
5 int main () {
6     int m=44, n = 66;
7     cout << "m = " << m << ", n = " << n << endl;
8     ++m;
9     --n;
10    cout << "m = " << m << ", n = " << n << endl;
11    m++;
12    n--;
13    cout << "m = " << m << ", n = " << n << endl;
14    cout << endl;
15    m=66;
16    n = ++m;
17    cout << "m = " << m << ", n = " << n << endl;
18    n=m++;
19    cout << "m = " << m << ", n = " << n << endl;
20    cout << "m = " << m++ << endl;
21    cout << "m = " << m << endl;
22    cout << "m = " << ++m << endl;
23    cout << "m = " << m << endl;
24
25    getch ();
26 }
```

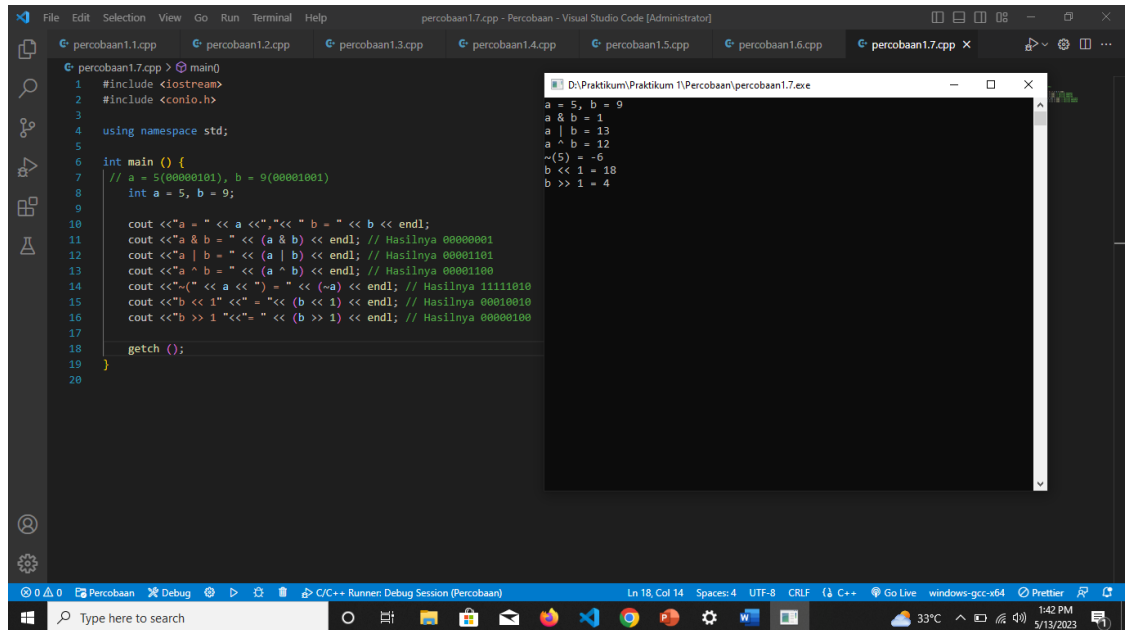
Output:

```
m = 44, n = 66
m = 45, n = 65
m = 46, n = 64

m = 67, n = 67
m = 68, n = 67
m = 69
m = 70
m = 70
```

Operator Increment dan Decrement

2.7 Percobaan 2.7



The screenshot shows a Visual Studio Code editor with a C++ file named `percobaan1.7.cpp`. The code defines two integers, `a = 5` and `b = 9`, and performs several bitwise operations. The output window displays the results of these operations, including binary representations and the results of AND, OR, XOR, and shift operations.

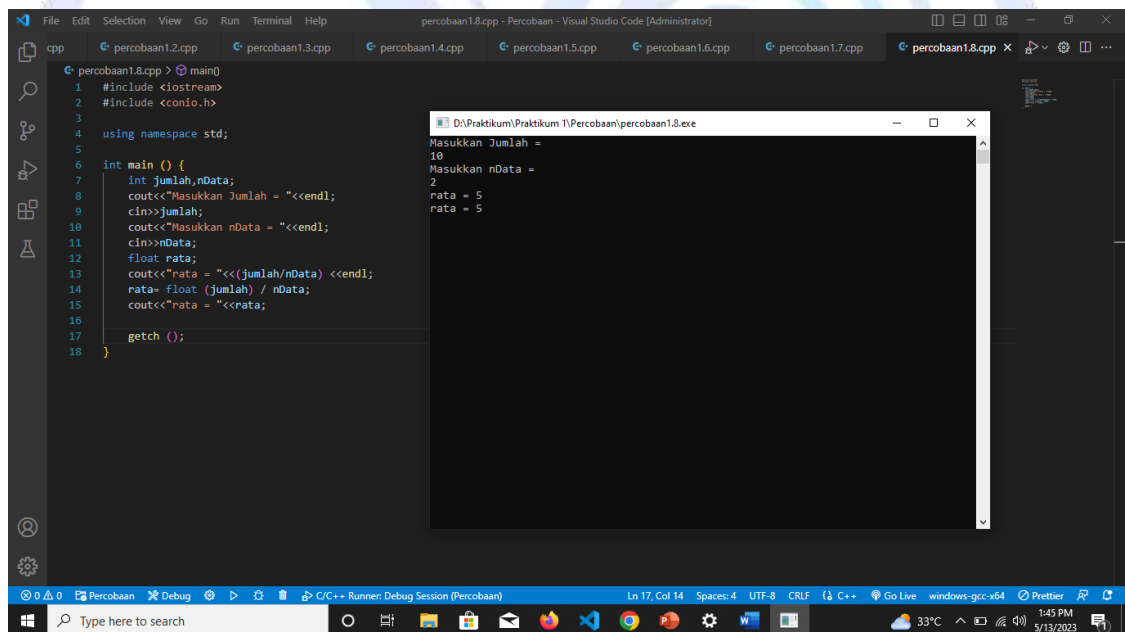
```
1 #include <iostream>
2 #include <conio.h>
3
4 using namespace std;
5
6 int main () {
7     // a = 5(00000101), b = 9(00001001)
8     int a = 5, b = 9;
9
10    cout <<"a = " << a <<"," <<" b = " << b << endl;
11    cout <<"a & b = " << (a & b) << endl; // Hasilnya 00000001
12    cout <<"a | b = " << (a | b) << endl; // Hasilnya 00001101
13    cout <<"a ^ b = " << (a ^ b) << endl; // Hasilnya 00001100
14    cout <<"~(a << 1) = " << (~a) << endl; // Hasilnya 11111010
15    cout <<"b << 1" << " = " << (b << 1) << endl; // Hasilnya 00010010
16    cout <<"b >> 1" << " = " << (b >> 1) << endl; // Hasilnya 00001000
17
18    getch ();
19 }
20
```

Output:

```
a = 5, b = 9
a & b = 1
a | b = 13
a ^ b = 12
~(5) = -6
b << 1 = 18
b >> 1 = 4
```

Operator Bitwise

2. 8 Percobaan 2.8



The screenshot shows a Visual Studio Code editor with a C++ file named `percobaan1.8.cpp`. The code prompts the user to input a sum and a number of data points, then calculates the average using type casting. The output window shows the user input and the calculated average.

```
1 #include <iostream>
2 #include <conio.h>
3
4 using namespace std;
5
6 int main () {
7     int jumlah,nData;
8     cout<<"Masukkan Jumlah = "<<endl;
9     cin>>jumlah;
10    cout<<"Masukkan nData = "<<endl;
11    cin>>nData;
12    float rata;
13    cout<<"rata = "<<((jumlah/nData) <<endl;
14    rata= Float (jumlah) / nData;
15    cout<<"rata = "<<rata;
16
17    getch ();
18 }

```

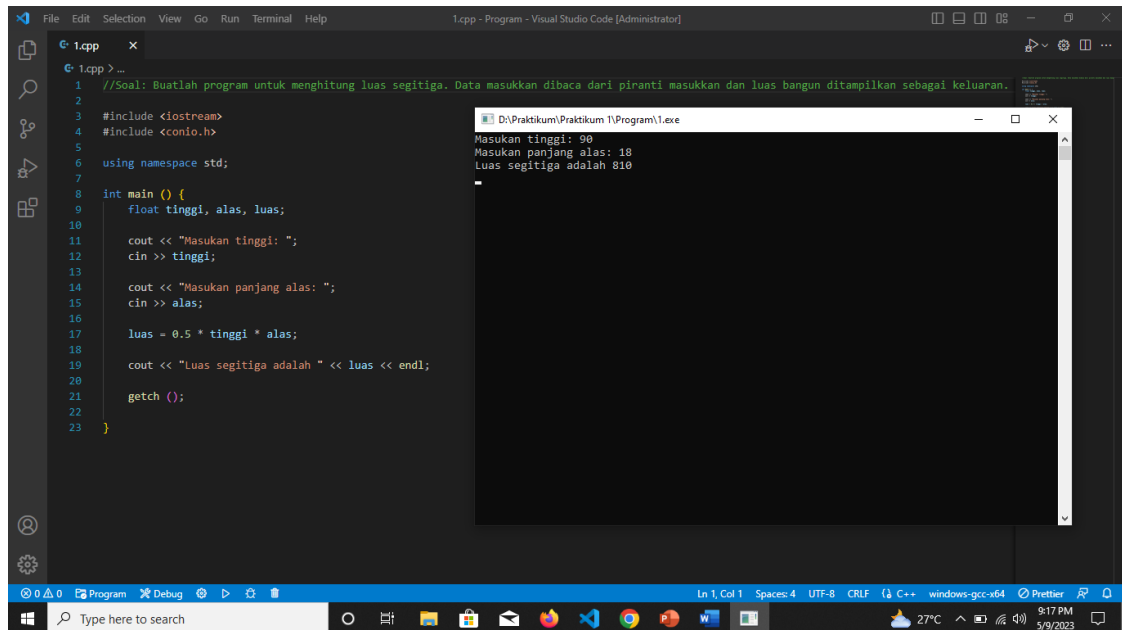
Output:

```
Masukkan Jumlah =
10
Masukkan nData =
2
rata = 5
rata = 5
```

Type Casting

BAB III TUGAS PRAKTIKUM

3.1 Tugas 1

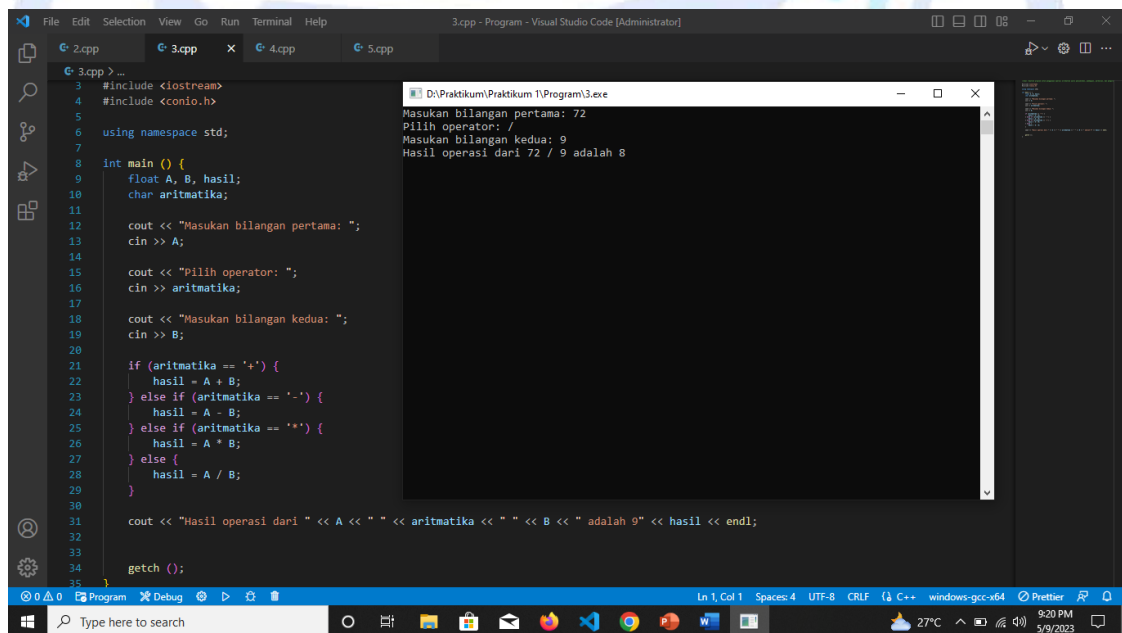


```
1 //Soal: Buatlah program untuk menghitung luas segitiga. Data masukan dibaca dari piranti masukan dan luas bangun ditampilkan sebagai keluaran.
2
3 #include <iostream>
4 #include <conio.h>
5
6 using namespace std;
7
8 int main () {
9     float tinggi, alas, luas;
10
11     cout << "Masukan tinggi: ";
12     cin >> tinggi;
13
14     cout << "Masukan panjang alas: ";
15     cin >> alas;
16
17     luas = 0.5 * tinggi * alas;
18
19     cout << "Luas segitiga adalah " << luas << endl;
20
21     getch ();
22 }
23
```

D:\Praktikum\Praktikum 1\Program\1.exe

Masukan tinggi: 90
Masukan panjang alas: 18
Luas segitiga adalah 810

3.2 Tugas 2

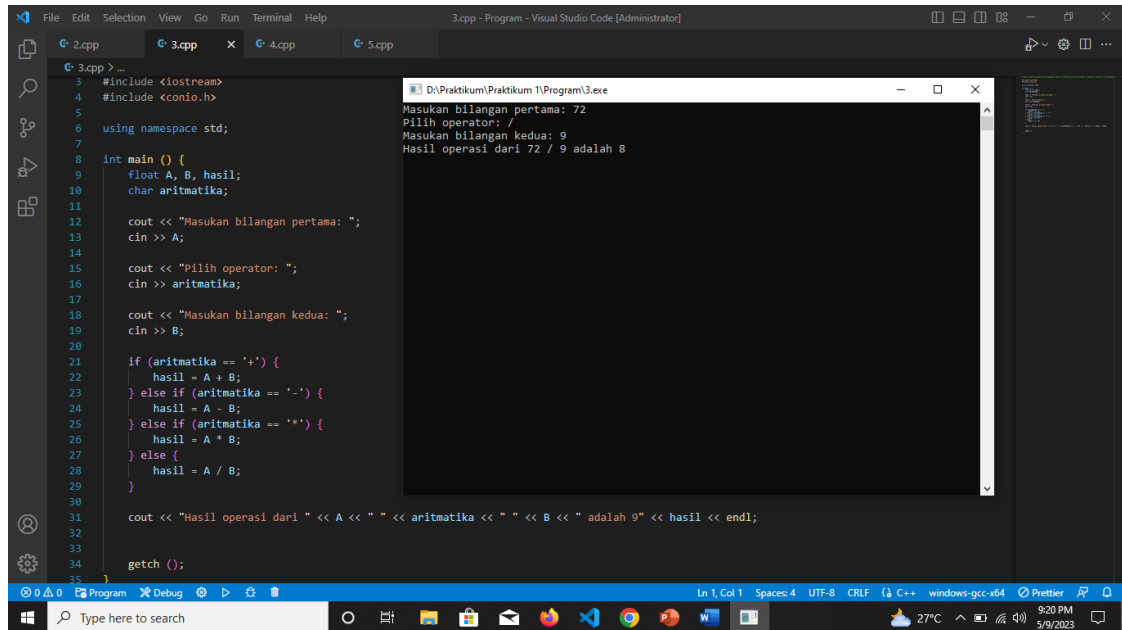


```
3 #include <iostream>
4 #include <conio.h>
5
6 using namespace std;
7
8 int main () {
9     float A, B, hasil;
10     char aritmatika;
11
12     cout << "Masukan bilangan pertama: ";
13     cin >> A;
14
15     cout << "Pilih operator: ";
16     cin >> aritmatika;
17
18     cout << "Masukan bilangan kedua: ";
19     cin >> B;
20
21     if (aritmatika == '+') {
22         hasil = A + B;
23     } else if (aritmatika == '-') {
24         hasil = A - B;
25     } else if (aritmatika == '*') {
26         hasil = A * B;
27     } else {
28         hasil = A / B;
29     }
30
31     cout << "Hasil operasi dari " << A << " " << aritmatika << " " << B << " adalah " << hasil << endl;
32
33     getch ();
34 }
35
```

D:\Praktikum\Praktikum 1\Program\3.exe

Masukan bilangan pertama: 72
Pilih operator: /
Masukan bilangan kedua: 9
Hasil operasi dari 72 / 9 adalah 8

3.3 Tugas 3

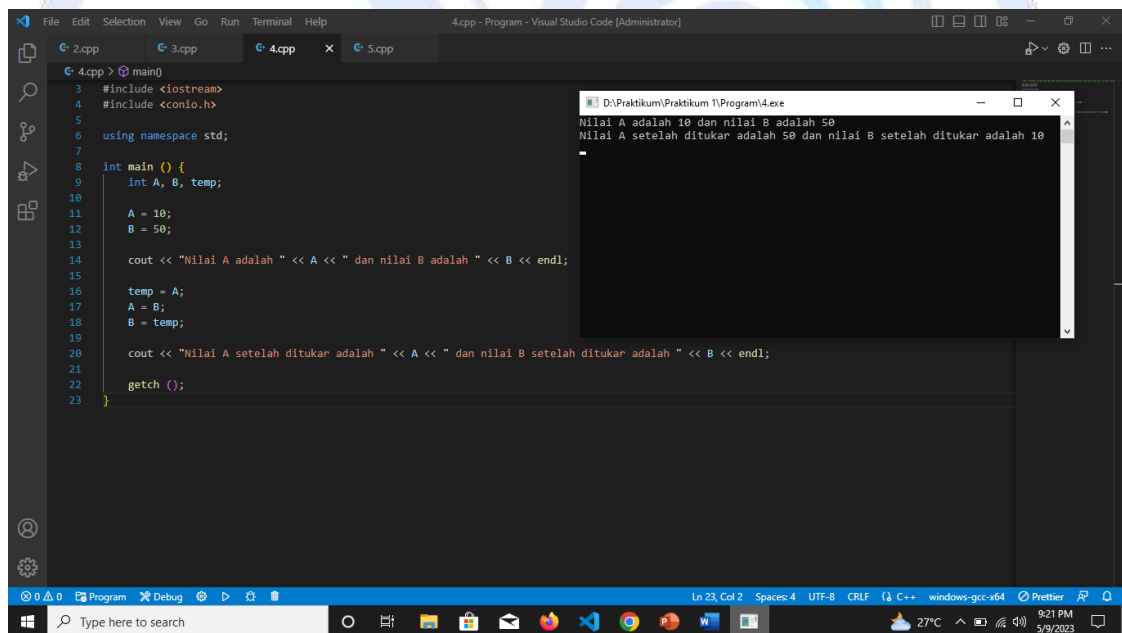


The screenshot shows a Visual Studio Code window with a C++ file named 3.cpp. The code implements a simple calculator that takes two numbers and an operator as input and outputs the result. The output window shows the program's execution with the following text:

```
Masukan bilangan pertama: 72
Pilih operator: /
Masukan bilangan kedua: 9
Hasil operasi dari 72 / 9 adalah 8
```

```
3.cpp >...
3 #include <iostream>
4 #include <conio.h>
5
6 using namespace std;
7
8 int main () {
9     float A, B, hasil;
10    char aritmatika;
11
12    cout << "Masukan bilangan pertama: ";
13    cin >> A;
14
15    cout << "Pilih operator: ";
16    cin >> aritmatika;
17
18    cout << "Masukan bilangan kedua: ";
19    cin >> B;
20
21    if (aritmatika == '+') {
22        hasil = A + B;
23    } else if (aritmatika == '-') {
24        hasil = A - B;
25    } else if (aritmatika == '*') {
26        hasil = A * B;
27    } else {
28        hasil = A / B;
29    }
30
31    cout << "Hasil operasi dari " << A << " * " << aritmatika << " * " << B << " adalah " << hasil << endl;
32
33    getch ();
34}
```

3.4 Tugas 4

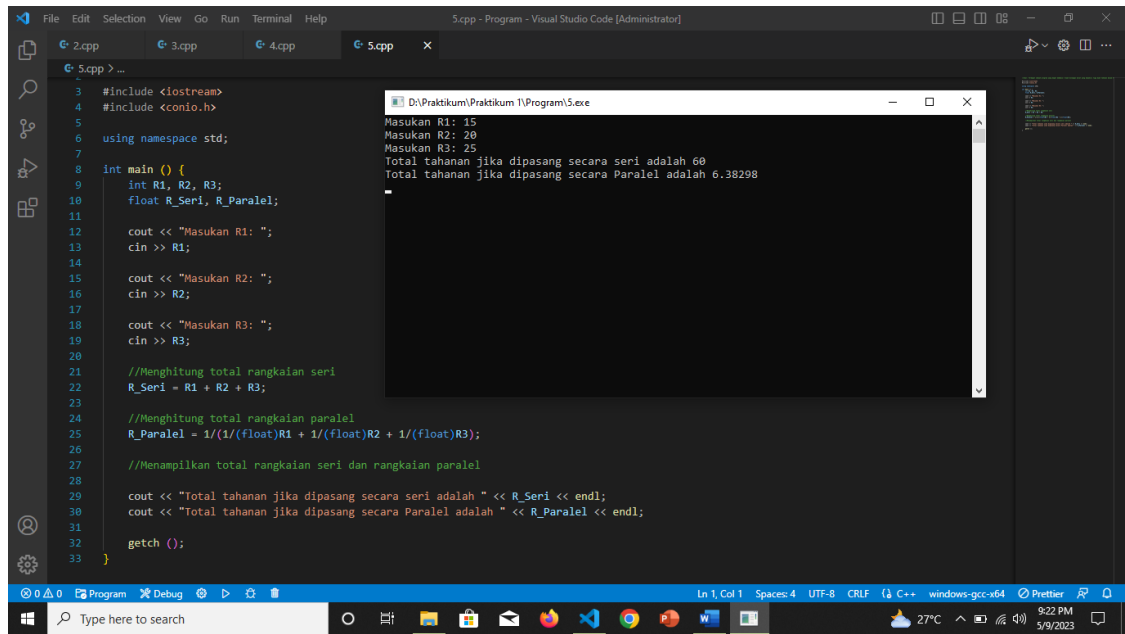


The screenshot shows a Visual Studio Code window with a C++ file named 4.cpp. The code implements a program that swaps the values of two variables, A and B, using a temporary variable. The output window shows the program's execution with the following text:

```
Nilai A adalah 10 dan nilai B adalah 50
Nilai A setelah ditukar adalah 50 dan nilai B setelah ditukar adalah 10
```

```
4.cpp > main0
3 #include <iostream>
4 #include <conio.h>
5
6 using namespace std;
7
8 int main () {
9     int A, B, temp;
10
11    A = 10;
12    B = 50;
13
14    cout << "Nilai A adalah " << A << " dan nilai B adalah " << B << endl;
15
16    temp = A;
17    A = B;
18    B = temp;
19
20    cout << "Nilai A setelah ditukar adalah " << A << " dan nilai B setelah ditukar adalah " << B << endl;
21
22    getch ();
23}
```

3.5 Tugas 5



```
5.cpp > ...
3 #include <iostream>
4 #include <conio.h>
5
6 using namespace std;
7
8 int main () {
9     int R1, R2, R3;
10    float R_Seri, R_Paralel;
11
12    cout << "Masukan R1: ";
13    cin >> R1;
14
15    cout << "Masukan R2: ";
16    cin >> R2;
17
18    cout << "Masukan R3: ";
19    cin >> R3;
20
21    //Menghitung total rangkaian seri
22    R_Seri = R1 + R2 + R3;
23
24    //Menghitung total rangkaian paralel
25    R_Paralel = 1/(1/(float)R1 + 1/(float)R2 + 1/(float)R3);
26
27    //Menampilkan total rangkaian seri dan rangkaian paralel
28
29    cout << "Total tahanan jika dipasang secara seri adalah " << R_Seri << endl;
30    cout << "Total tahanan jika dipasang secara Paralel adalah " << R_Paralel << endl;
31
32    getch ();
33 }
```

D:\Praktikum\Praktikum 1\Program\5.exe

Masukan R1: 15
Masukan R2: 20
Masukan R3: 25
Total tahanan jika dipasang secara seri adalah 60
Total tahanan jika dipasang secara Paralel adalah 6.38298



BAB IV ANALISA

4.1 Tugas 1

4.1.1 Poin A

Agar dapat menghitung dan menuliskan hasil dari luas segitiga, kita dapat menggunakan operator aritmatika bahasa C++. Dan untuk mencari luas segitiga, kita dapat menggunakan rumus : $a*t/2$ serta output yang akan kita keluarkan

“Luas Segitiga = ”(hasil kali a kali t bagi 2)

4.1.2 Poin B

Dengan menginputkan nilai alas dan tinggi pada program C++ yang telah kita buat, kita dapat mendapatkan hasil operasi aritmatika tersebut.

4.2 Tugas 2

4.2.1 Poin A

Untuk dapat menghitung dan mengeluarkan hasil dari total barang kita dapat menggunakan operator aritmatika bahasa C++. Dan untuk mencari harga total dari 5 barang, dapat menggunakan rumus : $5*harga_barang$ dengan output yang akan dihasilkan “Harga Total = ”(hasil perkalian 5 dengan harga barang)

4.2.2 Poin B

Dengan menginputkan nilai dari harga_barang, kita dapat menginputkan nilai dari harga barang ke rumus : $5*harga_barang$. Dapat juga mencari harga total dari 5 barang yang merupakan hasil operasi aritmatika tersebut.

4.3 Tugas 3

4.3.1 Poin A

Untuk dapat menentukan hasil dari operasi aritmatika, kita perlu mengetahui rumus aritmatika yang akan kita cari, ‘*’ sebagai kali, ‘/’ sebagai bagi, ‘+’ sebagai tambah dan ‘-’ sebagai kurang. Dengan menginputkan nilai $m=30$, $n=15$ dan $o=5$ kita bisa mengoperasikan dengan memakai rumus $m...n$ atau $m...o$ atau $n...o$. titik-titik tersebut nantinya bisa diisi dengan simbol kali, bagi, tambah, kurang . Output yang kita keluarkan adalah nilai

(nilai m)" + "(nilai n)" = " (hasil operasi $m+n$) ;dst

4.3.2 Poin B

Dengan menginputkan nilai m, n dan o yang diketahui kedalam rumus kita dapat mengetahui hasil operasi aritmatika menggunakan C++

4.4 Tugas 4

4.4.1 Poin A

Untuk dapat menggantikan letak nilai dan mengeluarkan hasil nilai yang akan digantikan, kita harus mengetahui nilai input yang akan diganti yaitu, A=10 dan B=50. kita bisa menggunakan variabel tambahan C=0 sehingga operasi atau proses algoritma yang kita buat C=A yang menghasilkan C=50, A=B yang menghasilkan A=50, B=C yang akan menghasilkan nilai A karena nilai C sebelumnya telah ditimpa nilai A yaitu B=10. Output yang dikeluarkan "A =" (nilai A) dan " ,B ="(nilai B)

4.4.2 Poin B

Dengan menginputkan nilai A, B dan C yang telah diketahui ke proses C=A;A=B;B=C kita dapat menggantikan nilai yang telah kita inputkan tadi dari A menjadi B dan B menjadi A

4.5 Tugas 5

4.5.1 Poin A

Untuk menentukan nilai resistor secara seri, paralel dan total, kita perlu memahami rumus fisika yang akan kita ubah ke dalam bentuk bahasa pemrograman C terlebih dahulu. Jika diketahui input nilai R1=2, R2 =3 dan R3=4 yang merupakan resistor kita bisa mencari nilai resistor dengan menggunakan rumus yaitu Resistor Seri = $R1+R2+R3$; 1/Resitor Paralel = $1/((1/R1)+(1/R2)+(1/R3))$ dan Resitor Total = Resistor Seri + Resitor Paralel. Output yang akan dikeluarkan " Resistor Seri ="(nilai total resistor seri), " Resitor Paralel ="(nilai total resistor paralel) dan " Resitor Total ="(nilai resistor total)

4.5.2 Poin B

Dengan menginputkan nilai resistor R1, R2 dan R3 pada rumus di poin A kita bisa mendapatkan total nilai resistor.

Dan saya menemukan masalah saat memasukan rumus resistor paralel yang seharusnya $1/\text{Resistor_Paralel}$ tidak bisa dipakai menggunakan float jadi saya memindahkan nilai $1/\text{Resistor_Paralel}$ di nilai Outputnya



BAB V PENUTUP

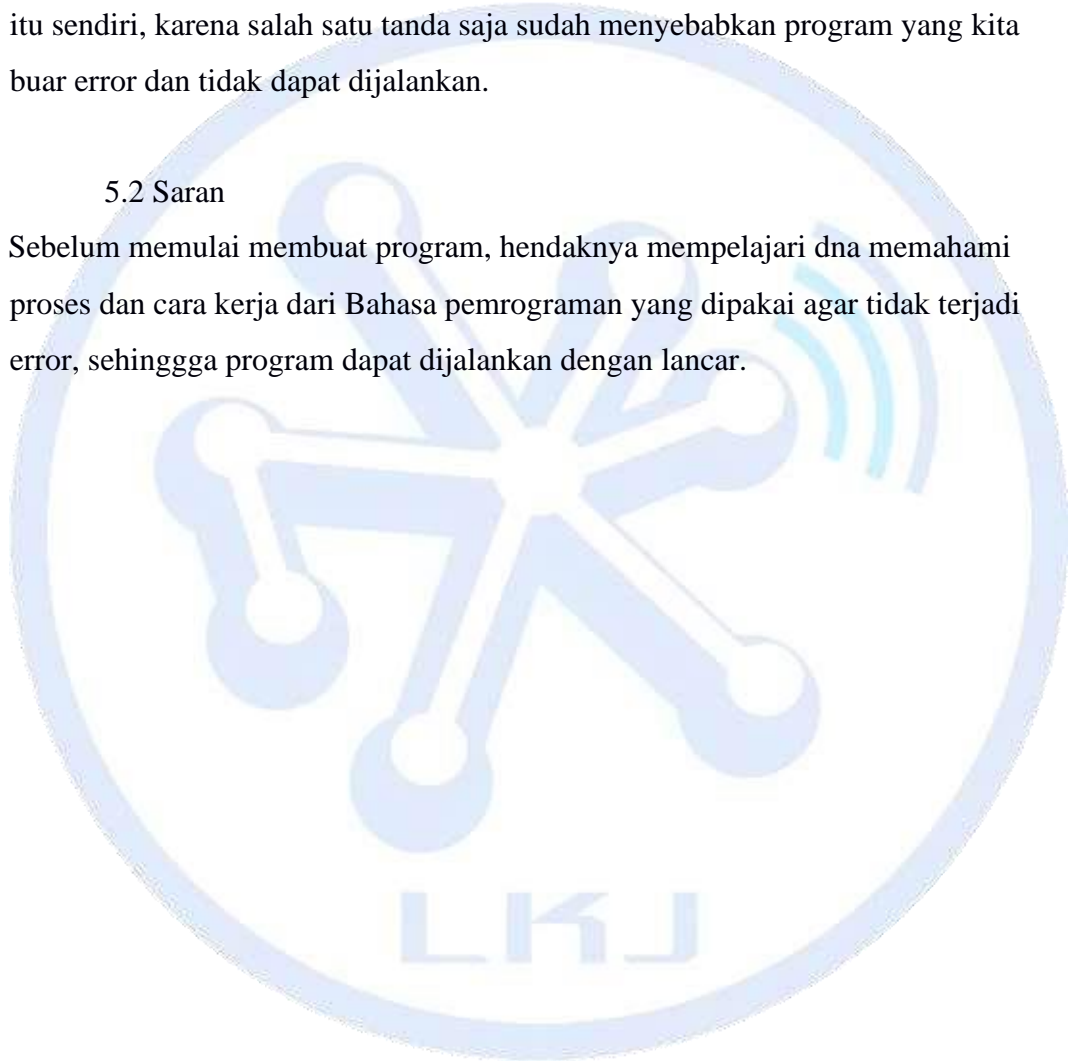
5.1 Kesimpulan

Sebelum membuat sebuah program, kita harus paham dan mengerti tentang system dari Bahasa pemrograman yang kita gunakan. Baik itu jenis, fungsi, perintah, dll. Dan untuk Bahasa C++ kali ini, kita harus memahami batang tubuh Bahasa tersebut.

Jadi dalam proses pemrograman, kita harus memahami proses dari pemrograman itu sendiri, karena salah satu tanda saja sudah menyebabkan program yang kita buar error dan tidak dapat dijalankan.

5.2 Saran

Sebelum memulai membuat program, hendaknya mempelajari dna memahami proses dan cara kerja dari Bahasa pemrograman yang dipakai agar tidak terjadi error, sehingga program dapat dijalankan dengan lancar.



DAFTAR PUSTAKA

- Anonim. Modul Praktikum Algoritma dan Pemrograman. Padang: Laboratorium Komputer dan Jaringan, Jurusan Teknik Komputer, 2023.
- Anonim. Modul Pengantar Bahasa Pemrograman C++. Padang: Laboratorium Komputer dan Jaringan, Jurusan Teknik Komputer, 2023
- Aritonang, Tsania Khairani. "PENGENALAN ALGORITMA PADA PEMBELAJARAN PEMROGRAMAN KOMPUTER." (2023).
- Joni,I Made,.,dan Budi Raharjo. Cara Mudah mempelajari Pemrograman C & Implementasinya. Bandung, 2006.

