

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-2



Disusun Oleh:

Nama : Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama : Andri Firman Saputra

Pratikum Algoritma

NIM : 201011402125

Laporan Awal - Pertemuan 2

Ringkasan Materi

a) Pendahuluan

Bahasa C++ merupakan bahasa pemrograman tingkat tinggi, yakni memakai kata-kata yg mudah dimengerti oleh manusia. Program C++ merupakan program yg berbentuk fungsi-fungsi. `main()` merupakan nama dari suatu fungsi yang harus ada di Program C++ dan di letakkan di bagian tertentu yang menunjukkan sebagai compiler dimana awal dari suatu program. Selain itu, `main()` hanya dapat digunakan sekali saja dalam satu program.

b-) Tipe Data

1) Tipe Data Integer

Tipe data integer atau bilangan bulat dapat dideklarasikan dgn menggunakan kata kunci (keyword) atau kata cadangan (reserved word), yaitu `short` atau `long`.

Apa juga tipe data integer yg dideklarasikan tidak beranda (`unsigned integer`) dgn menggunakan kata kunci `unsigned`.

2) Tipe Data Pecahan (Float)

Tipe data pecahan atau floating point dapat dideklarasikan dgn menggunakan kata kunci (keyword) atau kata cadangan (reserved), yaitu `float`, `double` dan `long double`.

c) Operator

Operator merupakan simbol atau kata yg dapat digunakan dalam program untuk melakukan suatu operasi atau manipulasi, seperti menjumlahkan dua buah nilai, memberikan nilai ke suatu variabel, membandingkan dua buah nilai dan lain-lain. Tanda operator dalam bahasa C/C++ dapat dikelompokkan sebagai berikut:

1) Operator Penyerjaan (Assignment Operator)

Operator penyerjaan / penugasan berupa tanda sama dengan (=) yang digunakan untuk mengisi nilai yang berada sebelah kanannya variabel yg ditunjukkan disebelah kiranya.

Contoh:

$a = 5$

$c = b = a$

mau nilai a dan b sama dengan c yaitu 5.

2) Operator aritmatika (arithmetic operator)

- Operator aritmatika yg tergolong sebagai operator binary.

*	Pengalian
/	Pembagian
%	Modulus atau sisa membagi
+	Penambahan
-	Pengurangan

- Operator aritmatika yg tergolong sebagai operator unary.

+	Tanda plus	$+x$
-	Tanda minus	$-x$
++	Prefix: Preincrement; Postfix: Postincrement	$x++$
--	Prefix: Predecrement; Postfix: Postdecrement	$x--$
$--x$ // Pre-decrement		
$++x$ // Pre-increment		
$x--$ // Post-decrement		
$x++$ // Post-increment		

3) Operator Peningkatan dan Penurunan (increment & decrement)

C/C++ mempunyai operator yg terkenal, yaitu operator Peningkatan (increment operator) yg menggunakan tanda Operasi ($++$). dan operator Penurunan yg menggunakan tanda Operasi ($--$). Operator penambahan digunakan untuk menambahkan nilai 1 pada nilai sebelumnya dan operator penurunan digunakan untuk mengurangi nilai 1 pada nilai sebelumnya.

4) Operator bitwise (bitwise operator)

Untuk operator bitwise dalam bentuk bit, bahasa C++ menyediakan beberapa operator seperti yg diperlihatkan pada tabel dibawah ini.

Operator	Keterangan
<<	Shift left (Pergeseran bit ke kiri)
>>	Shift right (Pergeseran bit ke kanan)
&	Operator bit AND
	Operator bit OR
^	Operator bit XOR
~	Operator bit NOT

5) Operator hubungan (relations operator)

Operator hubungan (relations operator) digunakan untuk membandingkan dua elemen nilai dan akan dihasilkan nilai perbandingannya, yaitu benar (bernilai 1) atau salah (bernilai 0). Operator ini banyak digunakan pada penyelesaian suatu kondisi di statement IF.

Operator	Keterangan
>	Lebih besar dari
>=	Lebih besar sama dengan
<	Lebih kecil dari
<=	Lebih kecil sama dengan
==	Sama dengan
!=	tidak sama dengan

6) Operator Logika

Operator logika biasanya berpasangan dgn operator hubungan. Operator logika dapat dilihat pada tabel berikut ini.

Operator	Keterangan
&&	Operator logika AND
	Operator logika OR
!	Operator logika NOT

- 7) Operator alamat (address operator)
C++ juga menyediakan dua buah operator alamat (address operator) yg berhubungan dgn penggunaan pointer.

Operator	Keterangan
&	Address of operator
*	Indirectio of operator

- 8) Operator koma (comma operator)
Operator koma (comma operator) digunakan untuk mengetahui beberapa ungkapan yg dipisahkan dgn tanda koma di dalam kurung buka dan kurung tutup.

- 9) Operator gabungan
C++ memungkinkan Penulisan suatu operator pengerjaan (=) untuk digabung dgn operator lainnya yaitu dgn operator aritmatika, operator peningkatan dan penurunan serta operator bitwise.

Bentuk penulisannya sebagai berikut:

Operator pengerjaan

Variabel = Variabel operator ungkapan

Dapat ditulis dengan tulisan sebagai berikut:

operator lainnya

operator pengerjaan

Variabel operator = Ungkapan

Nama : Andri Firman Saputra

Pratikum Algoritma

NIM : 201011402125

Laporan Awal 2 - Pertemuan 2

Tugas Pendahuluan

1. Tuliskan dan jelaskan kata-kata kunci pada bahasa C!
int = Untuk tipe data bilangan bulat pada Variabel
Float = Untuk tipe data bilangan pecahan pada Variabel
char = Untuk tipe data karakter pada Variabel
break = Untuk keluar dari Perulangan Switch()
2. Tuliskan dan jelaskan fungsi-fungsi yg digunakan pada bahasa C!
printf() = Untuk mencetak ke layar
scanf() = Untuk menginput data ke program
if() = Untuk membandingkan nilai (benar atau salah)
pow() = Untuk Pemanfaatan
3. Tuliskan dan jelaskan konstanta karakter escape (escape character) pada bahasa C!
\\a = bunyi bel/alert
\\b = Mundur 1 spasi ke belakang (backspace)
\\n = ganti baris baru
\\f = ganti halaman
4. Tuliskan dan jelaskan kode-kode format untuk fungsi printf()!
%d = int
%c = char
%f = float
%s = string

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-2



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir - Pertemuan 2



The image shows a C++ program in a code editor on the left and its execution output in a terminal window on the right.

Code Editor (Left):

```
1 // Pertemuan 2.cpp
2 #include <iostream>
3 using namespace std;
4
5 int main()
6 {
7     float jari, pi = 3.14, volume, luas;
8
9     cout << "Menghitung Volume dan Luas Permukaan Bola" << endl;
10
11     cout << "Masukkan nilai jari-jari: ";
12     int jari;
13
14     // Input volume [1]
15     // scanf("%d", &jari);
16     volume = (4 * pi * jari * jari * jari) / 3;
17
18     luas = 4 * pi * jari * jari;
19
20     cout << "Hasil volume bola adalah: " << volume << endl;
21     cout << "Hasil luas permukaan bola adalah: " << luas << endl;
22
23     return 0;
24 }
```

Terminal Window (Right):

```
W: D:\Programs\Dev-Cpp\bin\gcc.exe -o pertemuan2.exe
Menghitung Volume dan Luas Permukaan Bola
Masukkan nilai jari-jari: 30
Hasil volume bola adalah: 113040
Hasil luas permukaan bola adalah: 23544
Program selesai dalam 0.014 seconds with status value 0
Press any key to continue...
```

Kesimpulan – Pertemuan 2

Pada pertemuan ke-2 saya dapat menarik kesimpulan, dalam modul ini saya memahami tipe data pada pemrograman seperti: int, float, double. Selain itu, saya juga memahami operator – operator yaitu: operator penugasan, operator aritmatika (binary dan unary), operator bitwise, operator perbandingan, dan operator logika.

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-3



Disusun Oleh:

Nama : Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama : Andri Firman Sarvera

NIM : 201011402125

Pratikum Algoritma

Laporan Awal - Pertemuan 3

Teori Dasar

a) Pendahuluan

Perintah IF berguna untuk memilih dua atau lebih alternatif jawaban yg tersedia. Jika perintah IF tersebut terdiri dari dua atau lebih pernyataan, maka pernyataan tersebut harus berada di antara begin dan end.

Perintah IF memiliki bentuk umum:

```
IF (kondisi) {  
    pernyataan
```

```
}
```

Bentuk diatas mempunyai arti jika kondisi benar maka pernyataan dibawah dieksekusi.

Bisa juga seperti ini:

```
if (kondisi)
```

```
{
```

```
    pernyataan 1
```

```
    pernyataan 2
```

```
}
```

```
}
```

1) Perintah IF... ELSE

perintah ini mempunyai bentuk umum:

```
IF (kondisi) {
```

```
    pernyataan 1
```

```
} else {
```

```
    pernyataan 2
```

```
}
```

2) Perintah IF dalam IF

Perintah ini sering disebut nested-if. Bentuk umum:

```
if (kondisi) {
```

```
    if (kondisi) {
```

```
        pernyataan 1
```

```
    } else {
```

```
        pernyataan 2
```

```
    }
```

```
} else {
```

```
    pernyataan 3
```

```
}
```

Bentuk maksimum IF dalam IF Sering disebut IF bertingkat:

```
if (kondisi) {  
    Pernyataan 1  
} else if (kondisi 2) {  
    Pernyataan 2  
} else if (kondisi 3) {  
    Pernyataan 3  
} ... {  
} else {  
    Pernyataan  
}
```

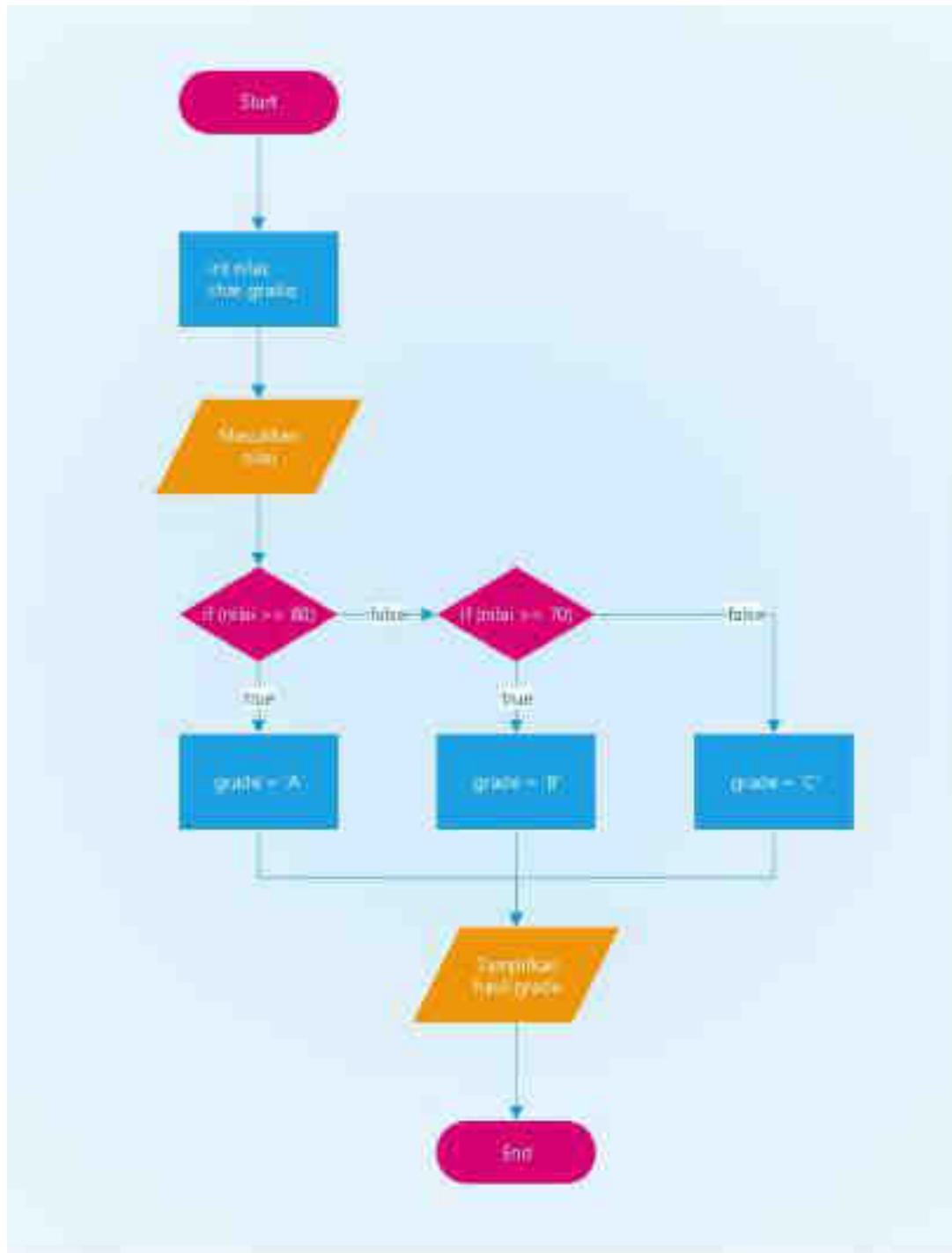
Tugas Pendahuluan

1. Jelaskan perbedaan perintah if dengan switch!
Perintah IF menggunakan kondisi boolean (benar/salah)
Sedangkan SWITCH menggunakan case dan mengerjakan
case berikutnya jika case sebelumnya tidak di break.
Dan switch hanya dapat dalam kondisi persamaan.
2. Sebutkan dan jelaskan jenis-jenis perintah IF!
IF (kondisi) : digunakan untuk mengambil keputusan,
Pernyataan ataupun pernyataan majemuk.
IF dalam IF : Pernyataan IF yg terletak di dalam IF.
Atau biasa disebut nested-if
3. mengapa anda sebagai programmer harus menggunakan
perintah dalam program!
karena, semua program membutuhkan pernyataan,
seperti keluar aksi, menghapus data atau bahkan
algoritma program.
4. Buatlah algoritma dan program sederhana menggunakan
perintah IF dgn memalui Flowchart!

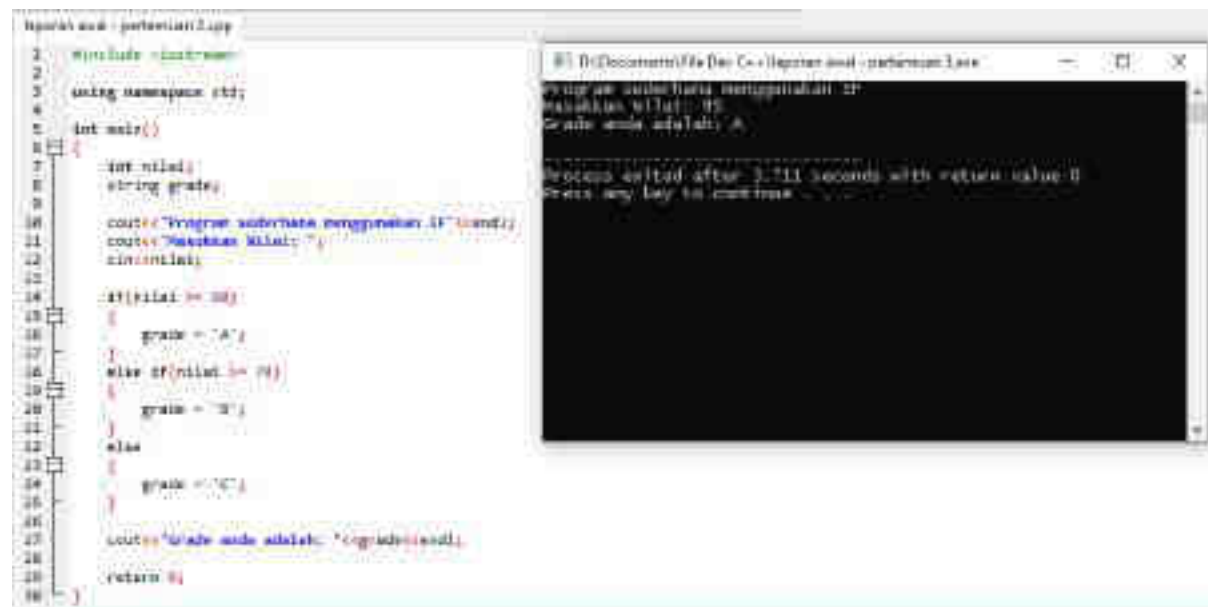
Algoritma:

1. Masukkan nilai
2. Melakukan pengkondisian
3. Menentukan grade dari hasil pengkondisian
4. Menampilkan grade

Flowchart



Program:



The image shows a C++ program in a code editor and its execution output in a terminal window.

Code Editor (Left):

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main()
6 {
7     int nilai;
8     string grade;
9
10    cout << "Program sederhana menggunakan if <endl>";
11    cout << "Masukkan Nilai: ";
12    cin >> nilai;
13
14    if (nilai >= 80)
15    {
16        grade = "A";
17    }
18    else if (nilai >= 70)
19    {
20        grade = "B";
21    }
22    else
23    {
24        grade = "C";
25    }
26
27    cout << "Grade anda adalah: " << grade << endl;
28    return 0;
29 }
```

Terminal Window (Right):

```
D:\Documents\file for C++\laporan soal - part 2\part 2.1.exe
Program sederhana menggunakan if
Masukkan nilai: 85
Grade anda adalah: A

Process exited after 3.711 seconds with return value 0
Press any key to continue . . .
```

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-3



Disusun Oleh:

Nama : Andri Firman Saputra

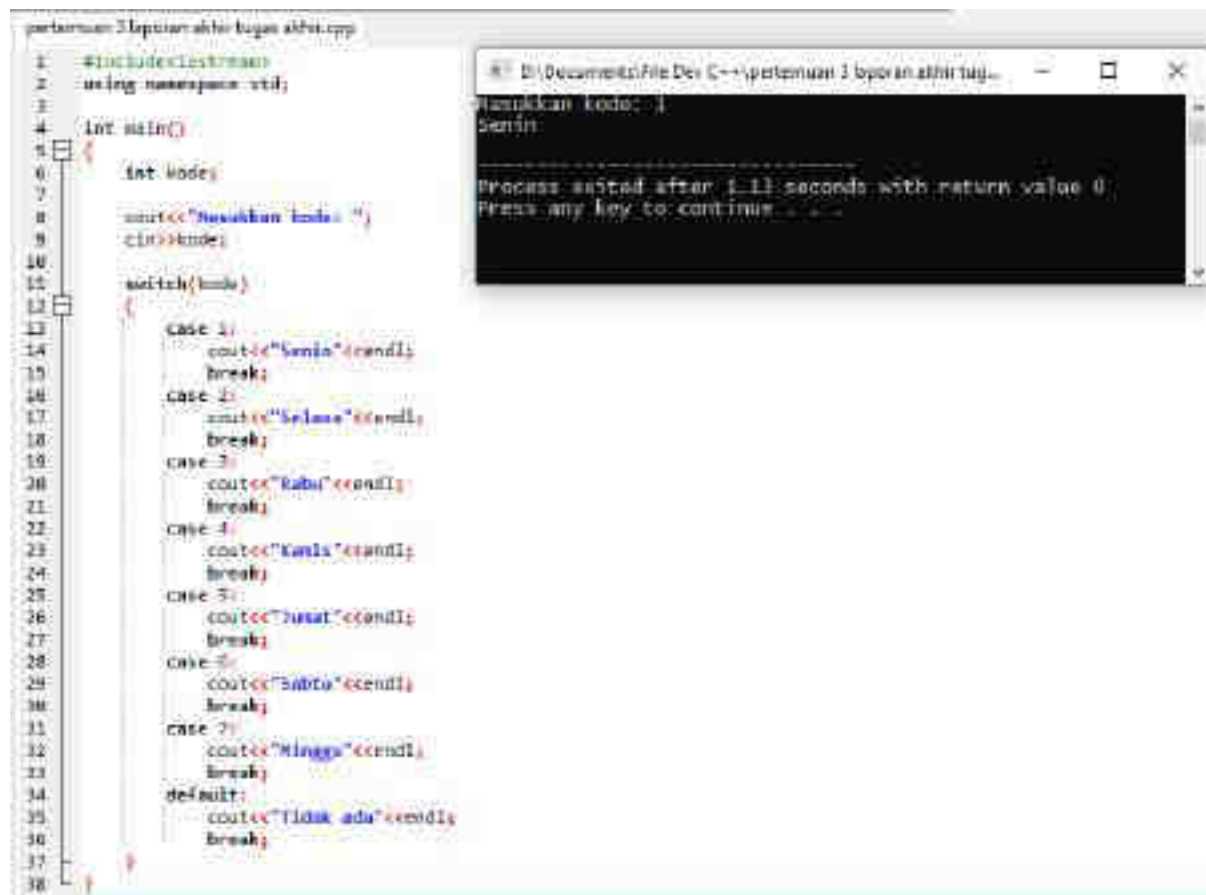
NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 3



The image shows a C++ IDE with a source code editor on the left and a console window on the right. The source code is a program that uses a switch statement to print the day of the week based on a numeric input. The console window shows the program's execution, including the prompt for input, the output of the day, and a message indicating the process ended after 1.11 seconds.

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     int kode;
7
8     cout << "Masukkan kode: ";
9     cin >> kode;
10
11     switch(kode)
12     {
13         case 1:
14             cout << "Senin" << endl;
15             break;
16         case 2:
17             cout << "Selasa" << endl;
18             break;
19         case 3:
20             cout << "Rabu" << endl;
21             break;
22         case 4:
23             cout << "Kamis" << endl;
24             break;
25         case 5:
26             cout << "Jumat" << endl;
27             break;
28         case 6:
29             cout << "Sabtu" << endl;
30             break;
31         case 7:
32             cout << "Minggu" << endl;
33             break;
34         default:
35             cout << "Tidak ada" << endl;
36             break;
37     }
38 }
```

Output from the console window:

```
D:\Documents\file Dev C++\pertemuan 1 bpw an akhir tugas...
Masukkan kode: 1
Senin
Process ended after 1.11 seconds with return value 0
Press any key to continue . . .
```

Source Code:

```
#include<iostream>
using namespace std;
int main()
{
    int kode;

    cout<<"Masukkan kode: ";
    cin>>kode;

    switch(kode)
    {
        case 1:
            cout<<"Senin"<<endl;
            break;
        case 2:
            cout<<"Selasa"<<endl;
            break;
        case 3:
            cout<<"Rabu"<<endl;
            break;
        case 4:
            cout<<"Kamis"<<endl;
            break;
        case 5:
            cout<<"Jumat"<<endl;
            break;
        case 6:
            cout<<"Sabtu"<<endl;
            break;
        case 7:
            cout<<"Minggu"<<endl;
            break;
        default:
            cout<<"Tidak ada"<<endl;
            break;
    }
}
```

Kesimpulan – Pertemuan 3

Pada pertemuan ke-3 ini, saya dapat menarik kesimpulan, saya memahami tentang *Decision*. Pada pemrograman ada dua yang saya ketahui, yaitu: if dan switch. Pada modul ini saya mempelajari: if biasa, if ... else, if majemuk, if dalam if (nested-if), dan if dalam if majemuk.

Contoh penggunaan-penggunaan if:

If biasa:

```
if (kondisi)
{
    pernyataan
}
```

If ... else:

```
if (kondisi 1)
{
    pernyataan 1
}
else
{
    pernyataan 2
}
```

If majemuk:

```
if (kondisi)
{
    pernyataan 1
    pernyataan 2
    pernyataan 3
    pernyataan n
}
```


If dalam if:

```
if (kondisi 1)
{
    if (kondisi 2)
    {
        pernyataan 1
    }

    pernyataan 2
}
```

If dalam if majemuk:

```
if (kondisi 1)
{
    pernyataan 1
}
else if (kondisi 2)
{
    pernyataan 2
}
else if (kondisi 3)
{
    pernyataan 3
}
else
{
    pernyataan n
}
```

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-4



Disusun Oleh:

Nama : Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama : Andri Firman Saputra

Praktikum Algoritma

NIM : 201011402125

Laporan Awal - Pertemuan 4

Teori Dasar

a) Pengalokasian

Perintah SWITCH merupakan pernyataan yg dirancang untuk menangani pengambilan keputusan yg melibatkan sejumlah pilihan alternatif yg di antarakan untuk menggantikan pernyataan IF bertingkat.

Bentuk Umum SWITCH
Switch (Eksresi)

{

Case konstanta 1;

Pernyataan 1;

break;

Case konstanta 2;

Pernyataan 2;

break;

default;

Pernyataan N;

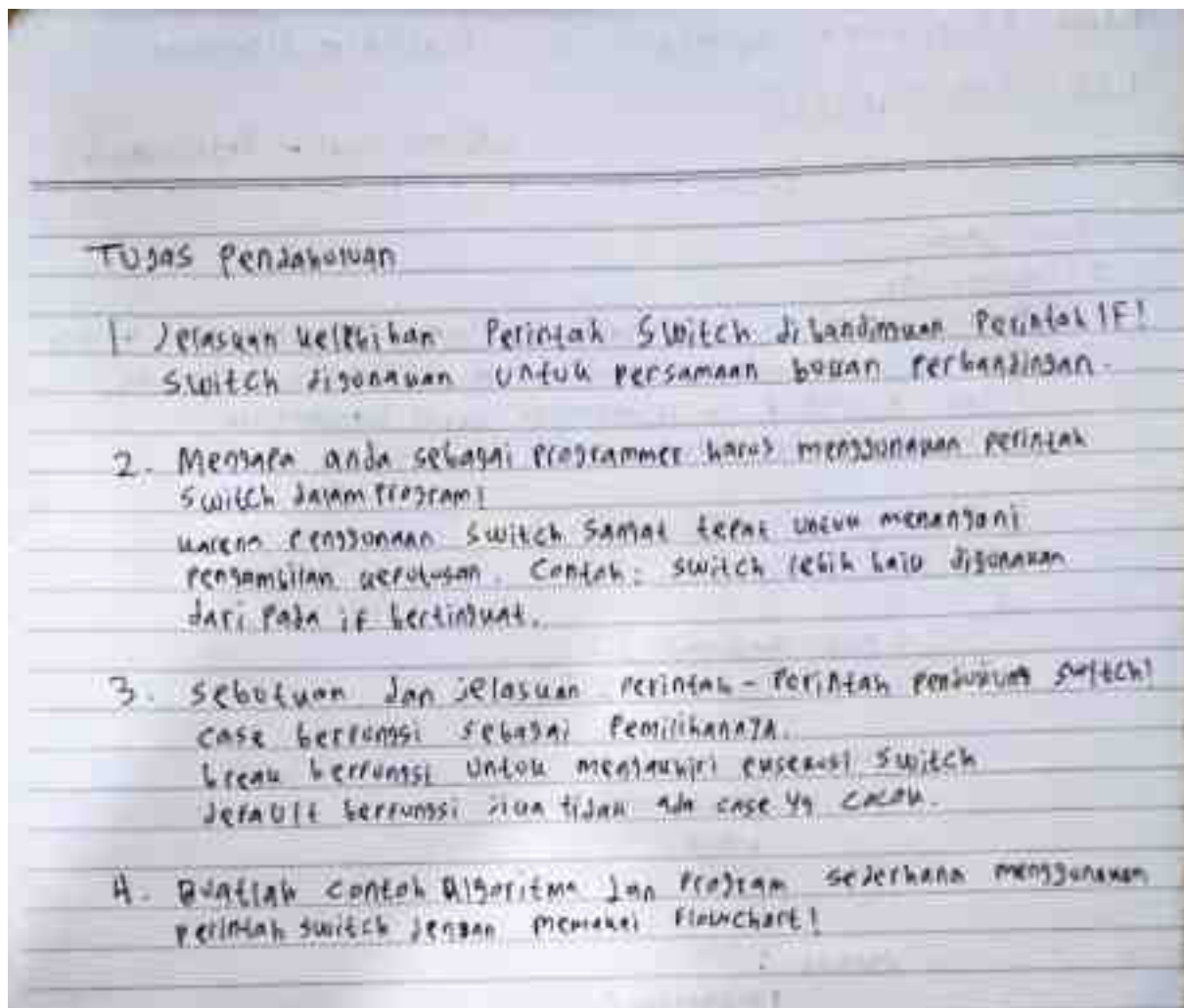
}

Eksresi bisa berupa integer atau char.

Setiap konstanta 1, konstanta 2, dst. dapat berupa konstanta integer atau konstanta char.

Setiap case dapat mempunyai satu atau lebih pernyataan. Pengujian pada switch dimulai dari case yg paling atas. Apabila nilai pada case cocok dgn ekspresinya maka akan dijalankan case tersebut. Tapi apabila tidak cocok maka akan bergeser ke case berikutnya. Sampai default jika ada.

Perintah break berfungsi untuk mengakhiri eksekusi. Bila tidak ada break case berikutnya diabaikan.

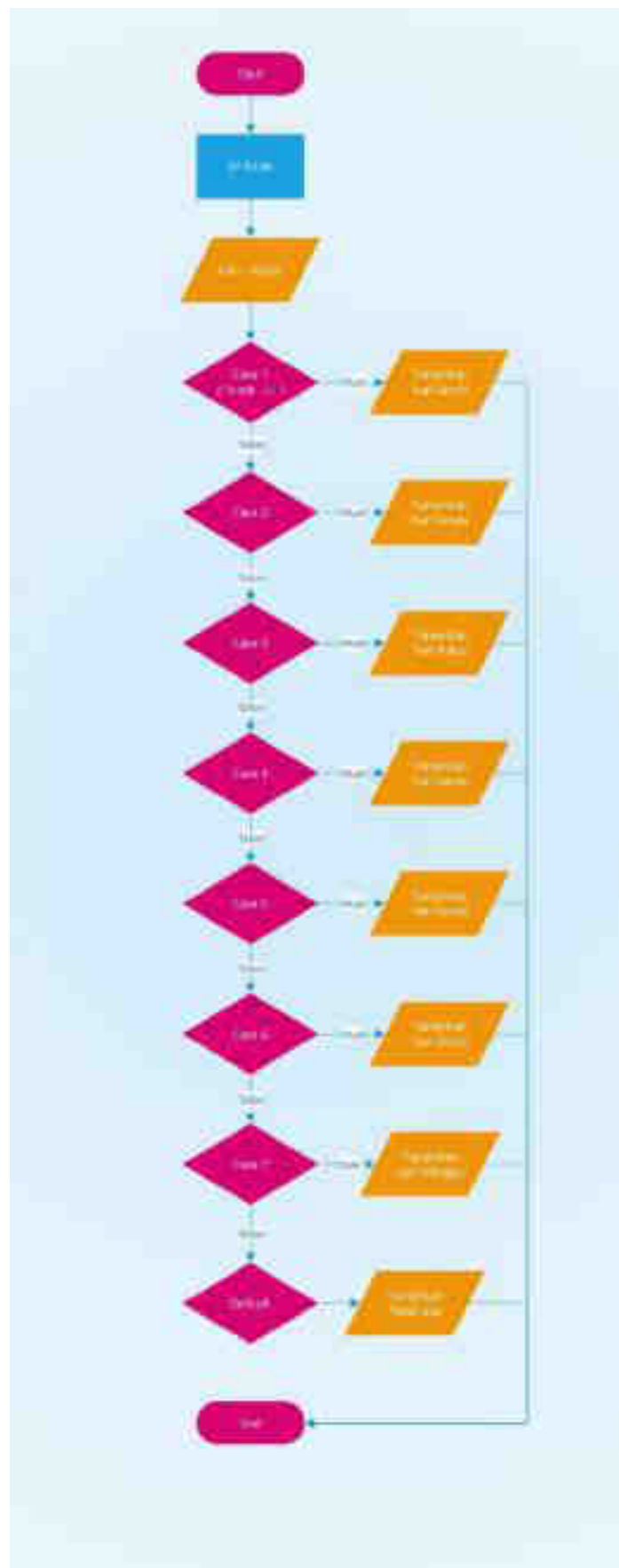


Jawaban No. 4

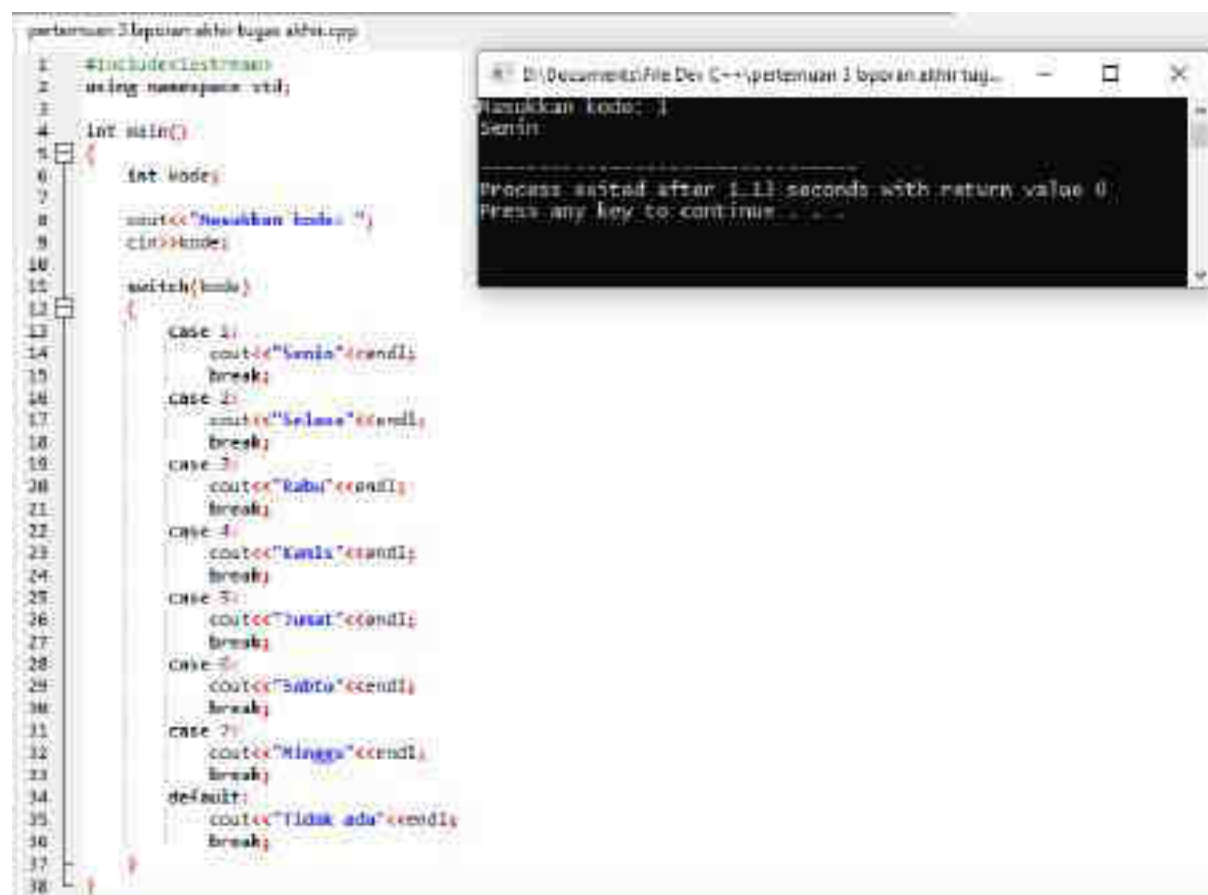
Algoritma:

1. Masukkan nilai kode
2. Switch melakukan pemilihan melalui case
3. Case mencocokkan nilai kode
4. Eksekusi pernyataan di dalam case yang nilainya cocok
5. Tampilkan isi case
6. Break, menghentikan eksekusi

Flowchart:



Contoh Program:



The image shows a C++ program in a code editor and its execution output in a terminal window.

Code Editor (perguruan 3 lopyan akhir tugas akhir.cpp):

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     int kode;
7
8     cout << "Masukkan kode: ";
9     cin >> kode;
10
11     switch(kode)
12     {
13         case 1:
14             cout << "Senin" << endl;
15             break;
16         case 2:
17             cout << "Selasa" << endl;
18             break;
19         case 3:
20             cout << "Rabu" << endl;
21             break;
22         case 4:
23             cout << "Kamis" << endl;
24             break;
25         case 5:
26             cout << "Jumat" << endl;
27             break;
28         case 6:
29             cout << "Sabtu" << endl;
30             break;
31         case 7:
32             cout << "Minggu" << endl;
33             break;
34         default:
35             cout << "Tidak ada" << endl;
36             break;
37     }
38 }
```

Terminal Window (D:\Documents\file Dev C++\perguruan 3 lopyan akhir tugas...):

```

Masukkan kode: 1
Senin
-----
Process exited after 1.11 seconds with return value 0
Press any key to continue . . .
```

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-4



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Petemuan 4

```

1 //ambil_harga_beli.cpp
2 using namespace std;
3
4 int main()
5 {
6     int harga_beli;
7
8     cout<<"Masukkan Harga Pembelian: ";
9     cin>>harga_beli;
10    if (harga_beli <= 100000)
11    {
12        cout<<"Diskon 50%";
13    }
14    else if (100000 < harga_beli <= 200000)
15    {
16        cout<<"Diskon 30%";
17    }
18    else if (200000 < harga_beli <= 300000)
19    {
20        cout<<"Diskon 20%";
21    }
22    else if (300000 < harga_beli <= 400000)
23    {
24        cout<<"Diskon 10%";
25    }
26    else if (400000 < harga_beli <= 500000)
27    {
28        cout<<"Diskon 5%";
29    }
30    else if (harga_beli > 500000)
31    {
32        cout<<"Diskon 0%";
33    }
34    else
35    {
36        cout<<"Error!";
37    }
38
39    return 0;
40 }

```

Output:

```

Masukkan Harga Pembelian: 100000
Diskon 50%

```


Source Code:

```
#include<iostream>
using namespace std;
int main ()
{
    int harga_beli;
    cout<<"Masukkan Harga Pembelian: ";
    cin>>harga_beli;
    if (100000 <= harga_beli && harga_beli < 200000)
    {
        cout<<"Discount 5%";
    }
    else if (200000 <= harga_beli && harga_beli < 300000)
    {
        cout<<"Tiket ke Yogya";
    }
    else if (300000 <= harga_beli && harga_beli < 400000)
    {
        cout<<"Tiket ke Bali";
    }
    else if (400000 <= harga_beli && harga_beli < 500000)
    {
        cout<<"Jam Tangan Rolex";
    }
    else if (harga_beli > 500000)
    {
        cout<<"Tiket ke Swiss";
    }
}
```

```
else
{
    cout<<"Coba Lagi"<<endl;
}
return 0;
}
```

Kesimpulan – Pertemuan 4

Pada pertemuan 4, saya mendapatkan kesimpulan, saya memahami tentang switch. Switch merupakan sintaks pemrograman untuk melakukan pemilihan dengan nilai persamaan, bukan dalam range. Switch akan menerima parameter yang digunakan untuk melakukan persamaan pada case, jika tidak ada yang memenuhi syarat dengan case, maka default akan dijalankan. Fungsi break untuk memberhentikan switch.

Bentuk umum switch:

```
switch(ekspresi)
{
    case konstanta1:
        pernyataan1;
        break;
    case konstanta2:
        pernyataan2;
        break;
    case konstanta_n;
        pernyataan_n;
        break;
    default:
        pernyataan_default;
}
```

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-5



Disusun Oleh:

Nama : Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama: Andri Firman Saputra
NIM: 201011402125

Praktikum Algoritma
Laporan Awal - Pertemuan 5

TEORI DASAR

a) Pendahuluan

Perintah pengulangan (looping) digunakan untuk melakukan proses berulang-ulang. Jika suatu kondisi sudah terpenuhi maka pelaksanaan akan dihentikan.

Proses pengulangan biasanya digunakan untuk proses pemasukan data, mengulang proses perhitungan dan mengulang proses penamplian hasil pengolahan data.

b) FOR

Proses looping yg menggunakan perulangan (counter) dapat dibuat dgn pernyataan FOR. Pernyataan ini digunakan bila anda sudah tahu berapa kali anda akan mengulang satu atau beberapa pernyataan.

Bentuk umum FOR:

```
for ( ungkapan 1 ; kondisi ; ungkapan 2 )
```

```
{
```

```
    Pernyataan ;
```

```
}
```

1. Ungkapan 1 adalah inisialisasi terhadap variabel penentu looping

2. kondisi adalah kondisi untuk keluar looping

3. Ungkapan 2 adalah pengaturan kenaikan atau penurunan nilai variabel penentu looping.

Tugas Penahuluan

1. Jelaskan kelebihan dan kekurangan perintah-perintah perulangan (looping) pada bahasa C/C++!

- kelebihan: Tidak perlu menggunakan statement dalam program hanya untuk menyelesaikan fungsi yg sama.
- kekurangan: Tidak dapat digunakan untuk menyelesaikan fungsi yg berbeda.

2. mengapa anda sebagai programmer harus menggunakan perintah for dalam program?

Karena for dalam program diperlukan dlm beberapa hal, misalkan mengulang proses memasukkan data, mengulang proses perhitungan, dan mengulang proses pemfian hasil pengolahan data.

3. Jelaskan perbedaan operator aritmatika $i++$ dan $++i$!

- $i++$: variabel akan dicetak terlebih dahulu kemudian ditambahkan
- $++i$: variabel akan ditambahkan terlebih dahulu kemudian dicetak.

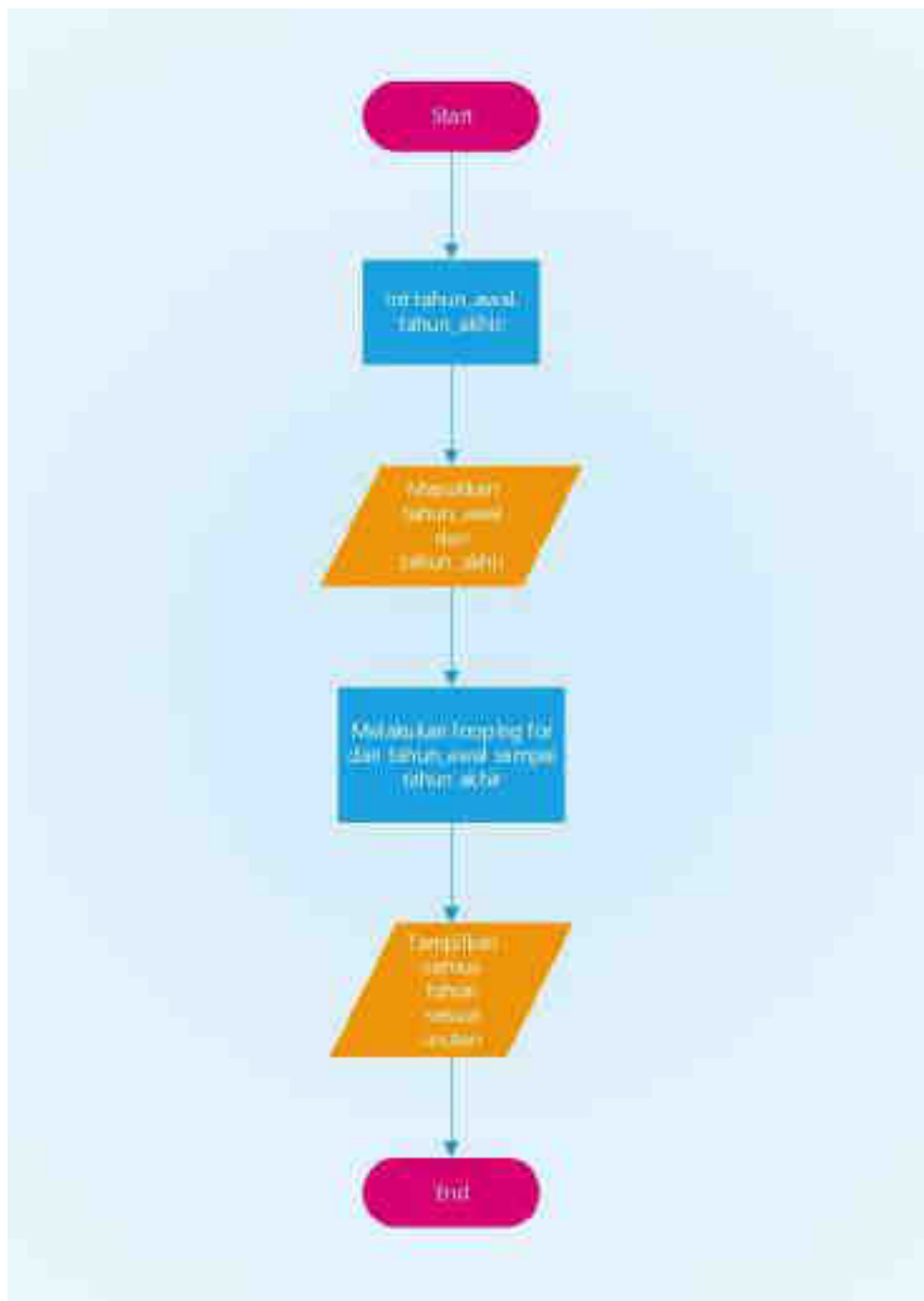
4. Buatlah contoh algoritma dan program sederhana menggunakan perintah for dan memakai flowchart!

Jawaban No. 4

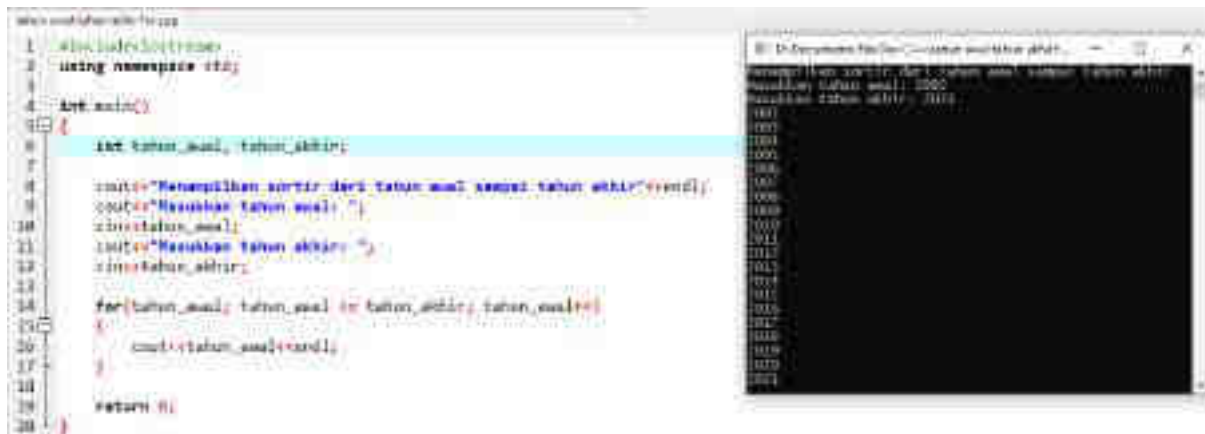
Algoritma

1. Masukkan nilai tahun awal
2. Masukkan nilai tahun akhir
3. Melakukan proses for
4. Tampilkan tahun
5. selesai

Flowchart



Contoh Program:



The image shows a screenshot of a C++ program in Visual Studio. The left pane displays the source code, and the right pane shows the program's output. The code defines a loop that prints the years from 1000 to 2000. The output window shows the years 1000 through 2000, each on a new line.

```
1 #include<iostream>
2 using namespace std;
3
4 int main()
5 {
6     int tahun_awal, tahun_akhir;
7
8     cout<<"Menampilkan sortir dari tahun awal sampai tahun akhir"<<endl;
9     cout<<"Masukkan tahun awal: ";
10    cin>>tahun_awal;
11    cout<<"Masukkan tahun akhir: ";
12    cin>>tahun_akhir;
13
14    for(tahun_awal; tahun_awal <= tahun_akhir; tahun_awal++)
15    {
16        cout<<tahun_awal<<endl;
17    }
18
19    return 0;
20 }
```

```
Menampilkan sortir dari tahun awal sampai tahun akhir
Masukkan tahun awal: 1000
Masukkan tahun akhir: 2000
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
```

Source code:

```
#include<iostream>
using namespace std;

int main ()
{
    int tahun_awal, tahun_akhir;

    cout<<"Menampilkan sortir dari tahun awal sampai tahun
akhir"<<endl;
    cout<<"Masukkan tahun awal: ";
    cin>>tahun_awal;
    cout<<"Masukkan tahun akhir: ";
    cin>>tahun_akhir;

    for (tahun_awal; tahun_awal <= tahun_akhir; tahun_awal++)
    {
        cout<<tahun_awal<<endl;
    }

    return 0;
}
```

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-5



Disusun Oleh:

Nama : Andri Firman Saputra

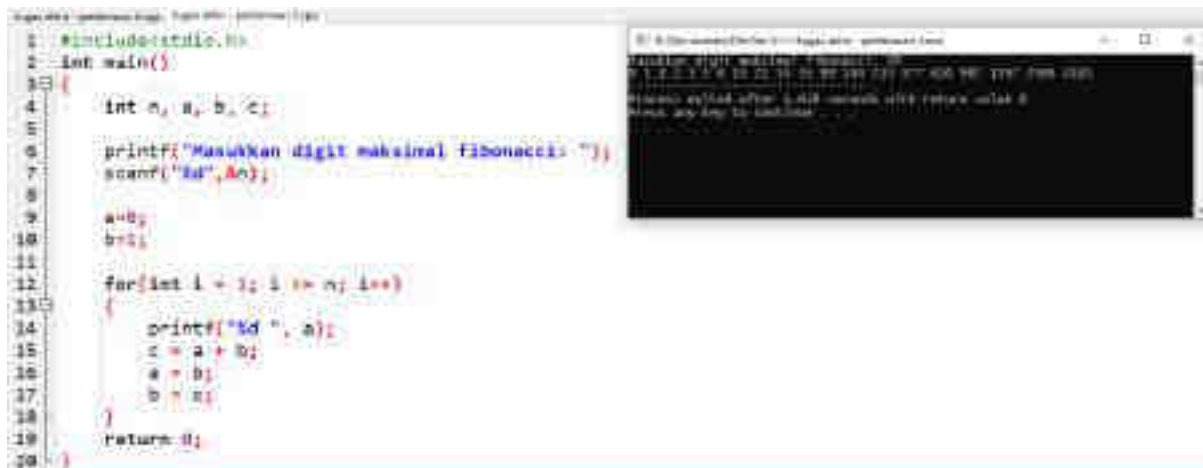
NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 5

The image shows a screenshot of a C program being edited in a text editor on the left and its execution output in a terminal window on the right. The code is a Fibonacci sequence generator. It prompts the user to enter the maximum digit of the Fibonacci sequence. The output shows the sequence of numbers generated for a given input. The code is as follows:

```
1 #include<stdio.h>
2 int main()
3 {
4     int n, a, b, c;
5
6     printf("Masukkan digit maksimal fibonacci: ");
7     scanf("%d",&n);
8
9     a=0;
10    b=1;
11
12    for(int i = 1; i <= n; i++)
13    {
14        printf("%d ", a);
15        c = a + b;
16        a = b;
17        b = c;
18    }
19    return 0;
20 }
```

The terminal output shows the user entering '10' and the program printing the first 10 Fibonacci numbers: 0 1 1 2 3 5 8 13 21 34.

Source Code:

```
#include<stdio.h>

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

    printf("Masukkan digit maksimal fibonacci: ");
    scanf("%d",&n);

    a=0;
    b=1;

    for(int i = 1; i <= n; i++)
    {
        printf("%d ", a);
        c = a + b;
        a = b;
        b = c;
    }
    return 0;
}
```


Kesimpulan

Pada pertemuan kali ini saya dapat menarik kesimpulan, saya memahami tentang *looping* menggunakan function for. fungsi looping berguna untuk melakukan hal yang sama (menginput data, menampilkan data, memanipulasi data) dengan sebuah fungsi looping dapat mencegah redudansi dan membuat program lebih dinamis serta mencegah pemborosan sintaks.

Bentuk umum for:

```
for (pernyataan 1; pernyataan 2; pernyataan 3) {  
    // hal yang ingin melakukan perulangan  
}
```

pernyataan 1 = untuk menginisialisasi variabel dan menentukan nilai awal

pernyataan 2 = untuk melakukan pengkodisian loop tetap dilakukan atau berhenti

pernyataan 3 = untuk menentukan increment atau decrement pada variabel di pernyataan 1

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-6



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama : Andri Firman Saputra

NIM : 201011402125

Praktikum Algoritma

Lataran Awal-Peremuan 6

Teori Dasar

a) Do... While

Perintah ini Menyakakan Pengulangan proses selama kondisi tertentu, mengerjakan do terlebih dahulu, kemudian melakukan Pengulangan pada while.

Bentuk Umum do While:

do

Pernyataan;

while (kondisi);

atau

do

{

pernyataan;

pernyataan;

}

while (kondisi);

b) While

Perintah while ini prinsipnya sama dgn perintah do...while hanya Pengulangan kondisinya terletak pada 100%. Langsung melakukan Pengulangan.

Bentuk Umum while:

while (kondisi)

pernyataan;

atau

while (kondisi)

{

pernyataan;

pernyataan;

}

pernyataan dapat berupa pernyataan tunggal atau beberapa pernyataan yg dilatasi dgn tanda {}. Pernyataan ini di jalankan bila kondisinya benar.

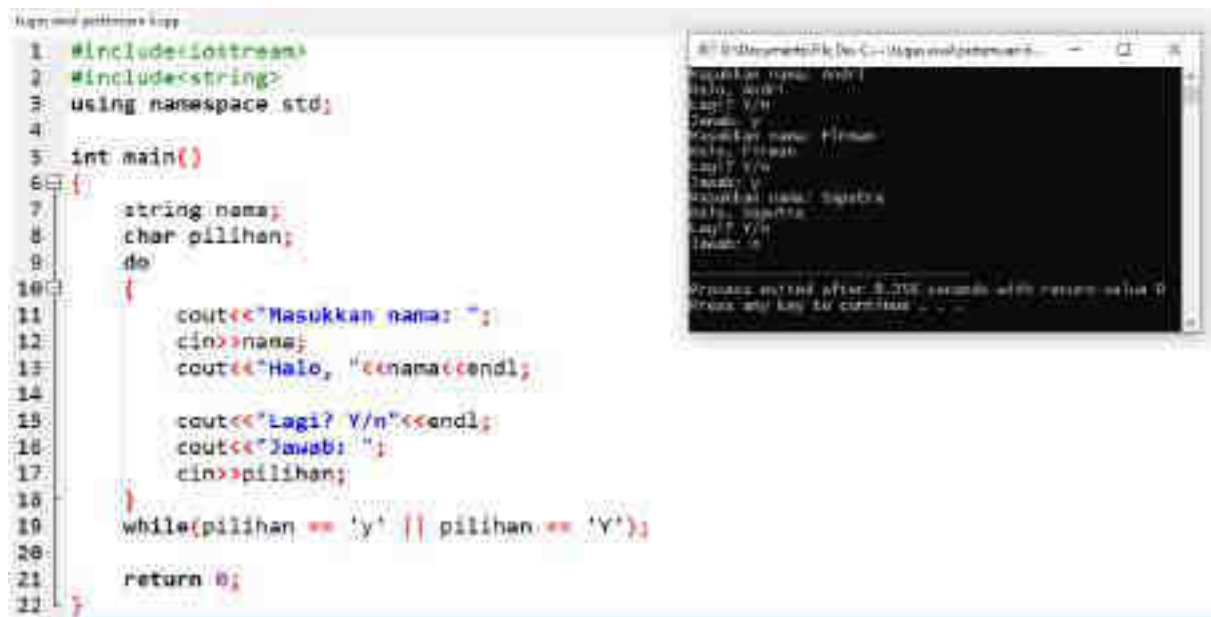
Tugas Pendahuluan

1. Mengapa anda selaku Programmer harus menggunakan perintah while atau do...while dalam program?
Karena untuk melakukan perintah pengulangan tidak perlu membuat kode secara berulang-ulang, cukup dgn menggunakan perintah while atau do while maka program akan mengulang sesuai dgn perintah.
2. Jelaskan kelebihan dan kekurangan perintah while dan do...while!
Kelebihan: Jarak mengulang kode program hanya harus menulis ulang.
Kekurangan: tidak bisa melakukan kode program yg berbeda.
3. Buatlah contoh Algoritma dan Program sederhana menggunakan perintah do...while dgn memalui Flowchart!
4. Buatlah contoh algoritma dan program sederhana menggunakan perintah while dgn memalui flowchart!

3. Algoritma

1. Masukkan nama
2. Tampilkan nama
3. Masukkan pilihan
4. Jika 'y', maka masukkan nama Kembali
5. Jika 'n', keluar program

Contoh program



The image shows a C++ program in a code editor and its execution output in a terminal window. The code is a simple loop that asks for a name and a choice to either repeat or exit. The terminal shows the program running with the name 'Hana' and the choice 'y' being entered, resulting in the name being printed again.

```
1 #include<iostream>
2 #include<string>
3 using namespace std;
4
5 int main()
6 {
7     string nama;
8     char pilihan;
9     do
10     {
11         cout<<"Masukkan nama: ";
12         cin>>nama;
13         cout<<"Halo, "<<nama<<endl;
14
15         cout<<"Lagi? Y/n"<<endl;
16         cout<<"Jawab: ";
17         cin>>pilihan;
18     }
19     while(pilihan == 'y' || pilihan == 'Y');
20
21     return 0;
22 }
```

```
Masukkan nama: hana
Halo, hana
Lagi? Y/n
Jawab: y
Masukkan nama: hana
Halo, hana
Lagi? Y/n
Jawab: n
Program ended after 0.002 seconds with return value 0
Press any key to continue . . .
```

Source Code:

```
#include<iostream>

#include<string>

using namespace std;

int main()

{

    string nama;

    char pilihan;
```



```
do
{
    cout<<"Masukkan nama: ";

    cin>>nama;

    cout<<"Halo, "<<nama<<endl;


    cout<<"Lagi? Y/n"<<endl;

    cout<<"Jawab: ";

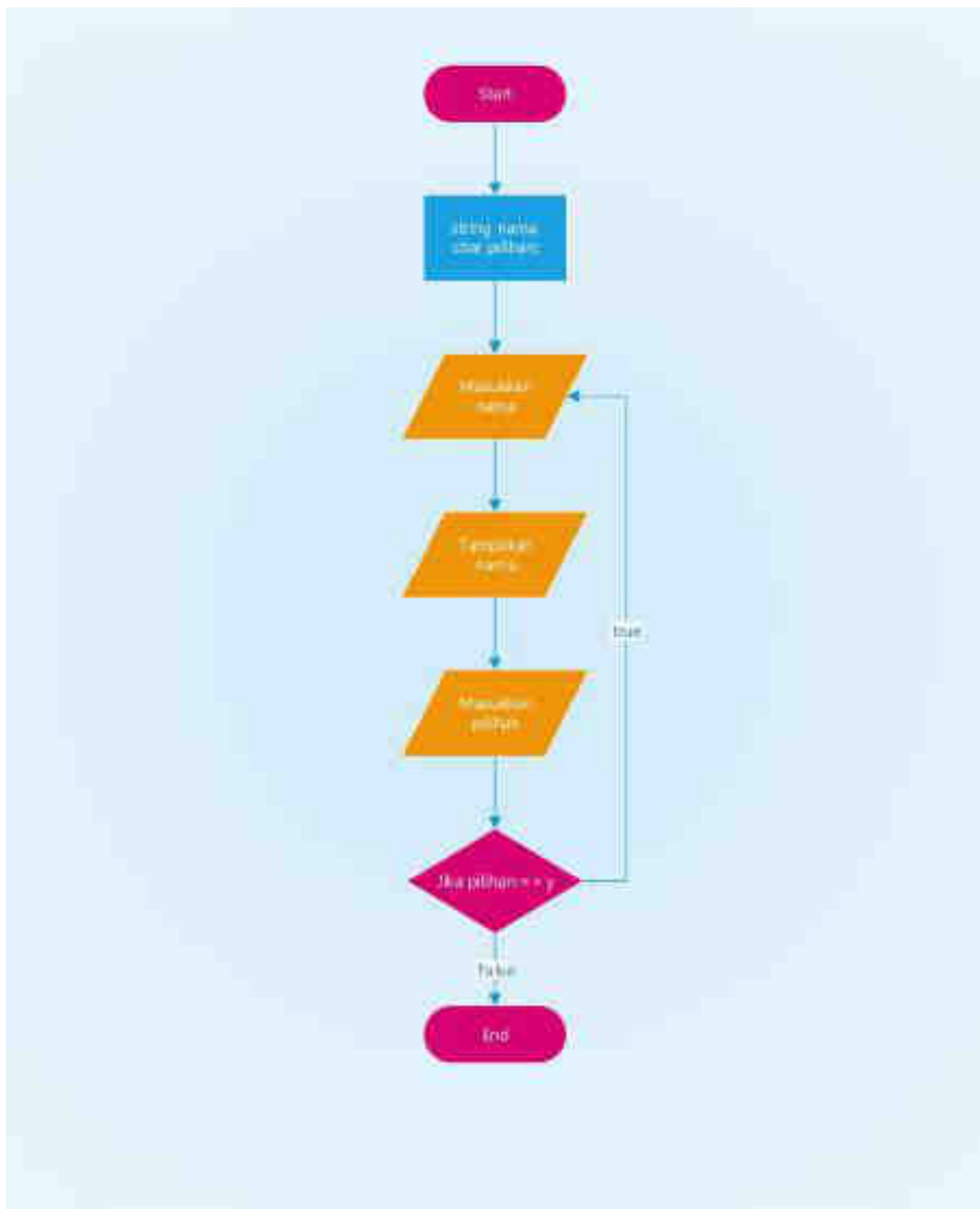
    cin>>pilihan;

}
while(pilihan == 'y' || pilihan == 'Y');

return 0;

}
```

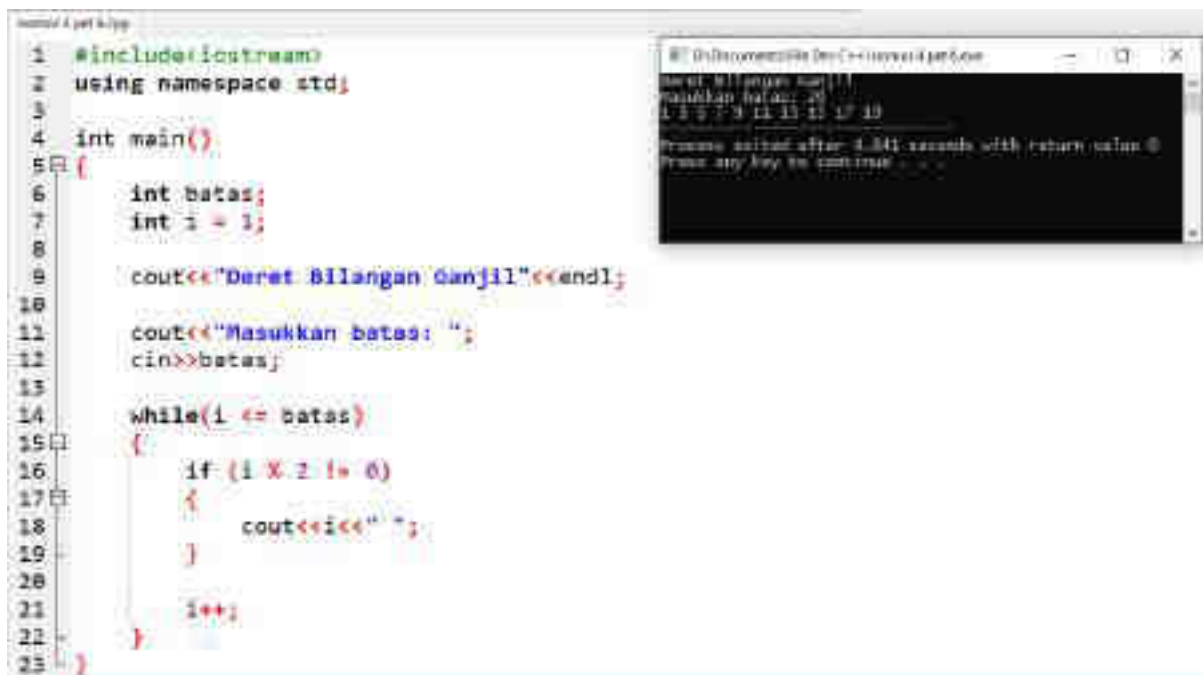
Flowchart



4. Algoritma

1. Masukkan batas maksimal nilai deret ganjil
2. Melakukan perulangan while dengan batas
3. Melakukan pengkondisian ganjil
4. Tampilkan deret ganjil

Contoh Program

The image shows a screenshot of a C++ program in a code editor and its execution output in a terminal window. The code is a program to print odd numbers up to a given limit. The code in the editor is as follows:

```
1 #include<iostream>
2 using namespace std;
3
4 int main()
5 {
6     int batas;
7     int i = 1;
8
9     cout<<"Deret Bilangan Ganjil"<<endl;
10
11     cout<<"Masukkan batas: ";
12     cin>>batas;
13
14     while(i <= batas)
15     {
16         if (i % 2 != 0)
17         {
18             cout<<i<<" ";
19         }
20
21         i++;
22     }
23 }
```

The terminal window shows the output of the program. It first prints "Deret Bilangan Ganjil" and then prompts the user to enter a limit. The user enters 20, and the program prints the sequence of odd numbers from 1 to 19: "1 3 5 7 9 11 13 15 17 19".

Source Code:

```
#include<iostream>

using namespace std;

int main()

{

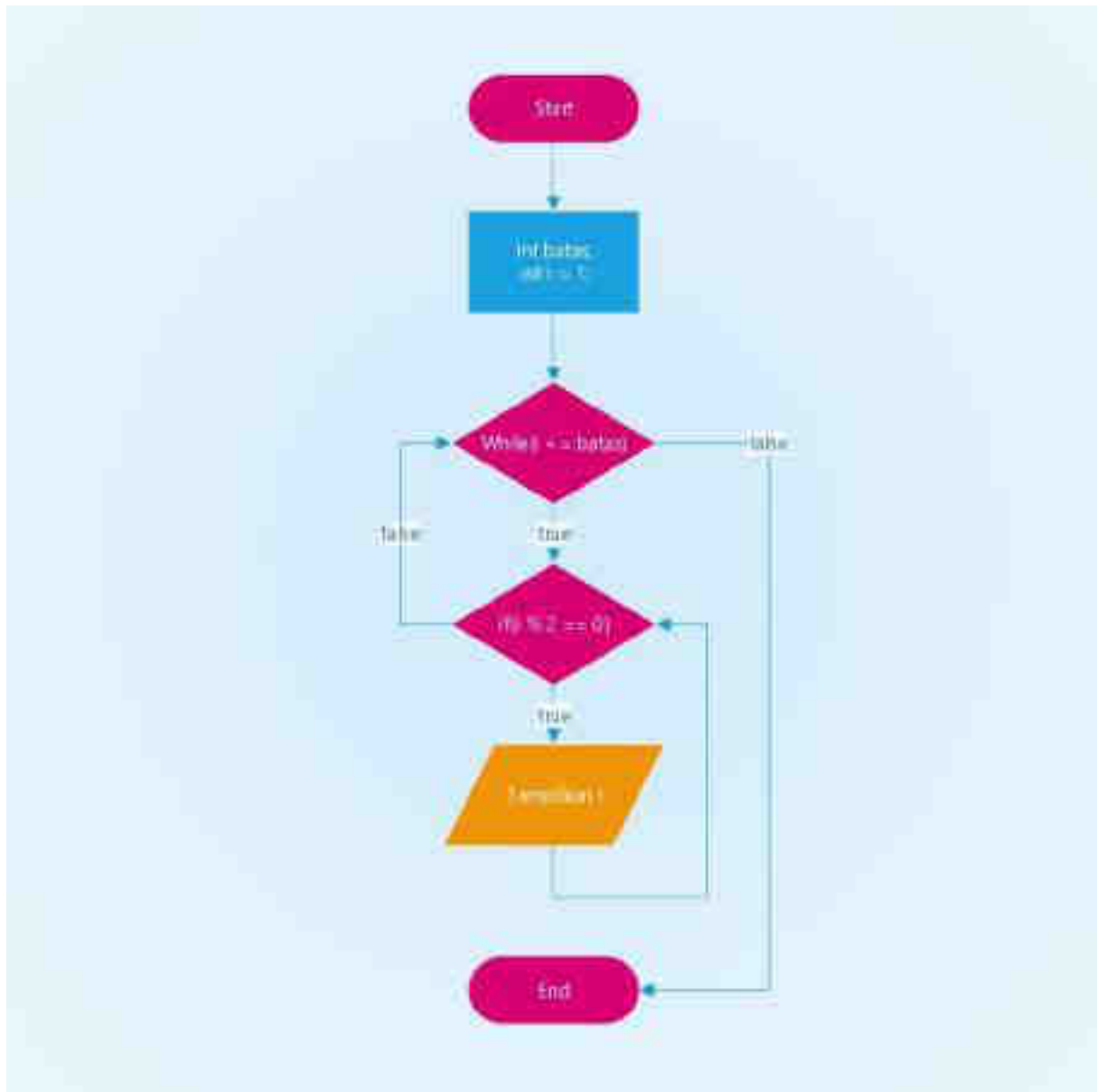
    int batas;

    int i = 1;

    cout<<"Deret Bilangan Ganjil"<<endl;
```

```
cout<<"Masukkan batas: ";  
cin>>batas;  
  
while(i <= batas)  
{  
    if (i % 2 != 0)  
    {  
        cout<<i<<" ";  
    }  
  
    i++;  
}  
}
```

Flowchart



LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-6



Disusun Oleh:

Nama: Andri Firman Saputra


NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 6



The screenshot shows a C++ program in a text editor and its execution in a terminal window.

Code:

```

1  #include<iostream>
2  using namespace std;
3
4  int main()
5  {
6      int n, i, a, b, c;
7
8      cout<<"Masukkan digit maksimal Fibonacci: ";
9      cin>>n;
10
11     i=1;
12     a=0;
13     b=1;
14
15     while(i<=n)
16     {
17         cout<<";
18         c = a + b;
19         a = b;
20         b = c;
21         i++;
22     }
23     return 0;
24 }

```

Output:

```

Masukkan digit maksimal Fibonacci: 15
0 1 1 2 3 5 8 13 21 34 55 89 144 233
Process exited after 1.092 seconds with return value 0
Press any key to continue . . .

```

Source Code:

```
#include<iostream>
using namespace std;

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

    cout<<"Masukkan digit maksimal fibonacci: ";
    cin>>n;

    i=1;
    a=0;
    b=1;

    while(i<=n)
    {
        cout<<a;
        c = a + b;
        a = b;
        b = c;
        i++;
    }
    return 0;
}
```

Kesimpulan

Pada modul ke 6 ini saya bisa menarik kesimpulan, Saya memahami konsep perulangan menggunakan while dan do ... while.

Bentuk umum while:

```
while(keadaan)
{
    pernyataan1;
    pernyataan2;
    pernyataan_n;
}
```

Bentuk umum do ... while:

```
do
{
    pernyataan1;
    pernyataan2;
    pernyataan_n;
}
while(keadaan);
```

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-7



Disusun Oleh:

Nama : Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama : Andri Firman Sarutra

Praktikum Algoritma

NIM : 201011402125

Laporan Awal - Pertemuan 7

Teori Dasar

a) Dasar Fungsi

Pada umumnya fungsi memerlukan masukan yg dinamakan argument atau parameter. Hasil akhir fungsi akan berupa sebuah nilai (nilai fungsi lain)

Bentuk Umum Fungsi:

```
penentu_tipe nama_fungsi (daftar parameter)
```

```
{
```

```
    deklarasi parameter
```

```
    tubuh fungsi
```

```
}
```

Penentu tipe berfungsi untuk menentukan tipe keluaran fungsi yg dapat berupa salah satu data C++ yg berlaku, seperti char, int, default tipe fungsi yg tidak disebutkan dianggap sbg int.

Sebuah fungsi dapat tidak mempunyai parameter

contoh:

```
inisialisasi()
```

```
{
```

```
    return;
```

```
}
```

Penggunaan fungsi:

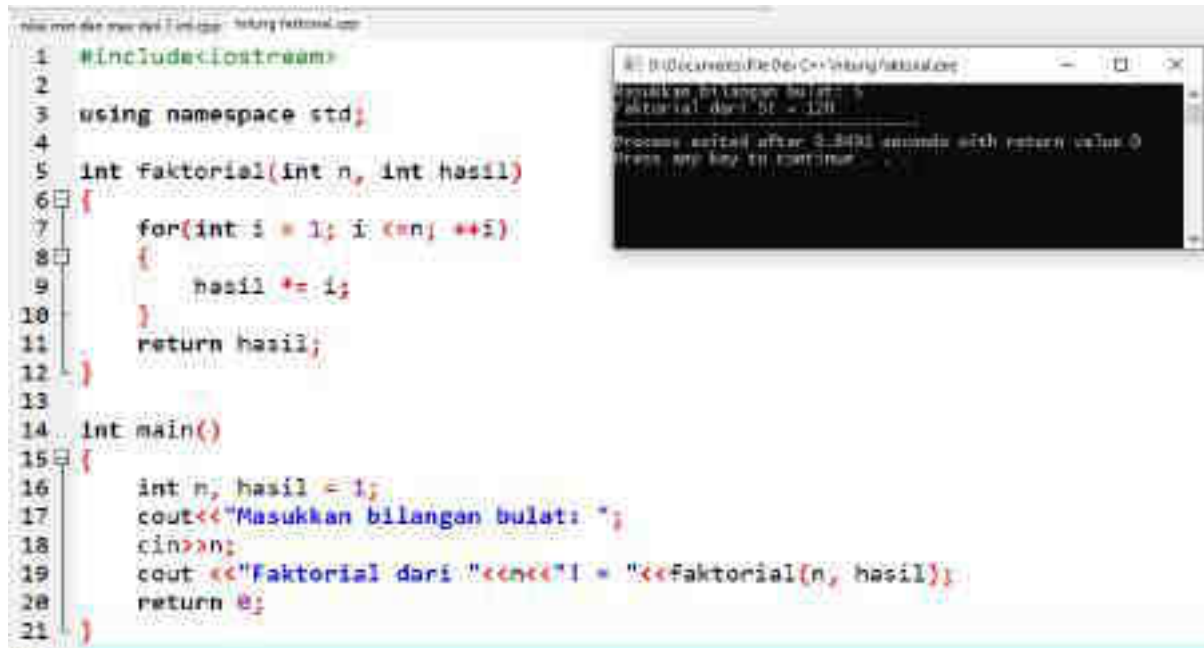
```
nama_fungsi (parameter jika ada);
```


Tugas Pendahuluan

1. Jelaskan yg dimaksud dgn rekursi pada Fungsi pada bahasa Pemrograman C/C++!
fungsi yg memanggil dirinya sendiri secara langsung ataupun tidak dan proses pemanggilannya disebut rekursi.
2. Sebutkan dan jelaskan Perintah - Perintah pendukung fungsi!
 - tipe_keluaran = jenis tipe data yg di keluarkan / dikembalikan. seperti : int, float, char, string dan lain-lain. Selain itu, dapat juga tidak mengembalikan nilai yaitu, void
 - parameter = argumen untuk fungsi yg memiliki tipe data. sebuah fungsi boleh atau tidak ada parameter.
3. Jelaskan kelebihan menggunakan fungsi pada bahasa Pemrograman C/C++?
 - memudahkan dalam pengembangan program
 - reusable (dapat digunakan kembali)
 - Programnya dapat dibagi beberapa sub-sub program.
4. Buatlah contoh algoritma dan program sederhana menggunakan fungsi dengan memalui flowchart!

Jawaban No. 4

Contoh Program



The image shows a C++ program in a code editor and its execution output in a console window. The code defines a recursive function 'faktorial' and a 'main' function that takes user input and calls the function. The console output shows the program running, taking input '5', and displaying the result 'Faktorial dari 5 = 120'.

```
1 #include<iostream>
2
3 using namespace std;
4
5 int faktorial(int n, int hasil)
6 {
7     for(int i = 1; i <= n; ++i)
8     {
9         hasil *= i;
10    }
11    return hasil;
12 }
13
14 int main()
15 {
16     int n, hasil = 1;
17     cout<<"Masukkan bilangan bulat: ";
18     cin>>n;
19     cout <<"Faktorial dari "<<n<<"! = "<<faktorial(n, hasil);
20     return 0;
21 }
```

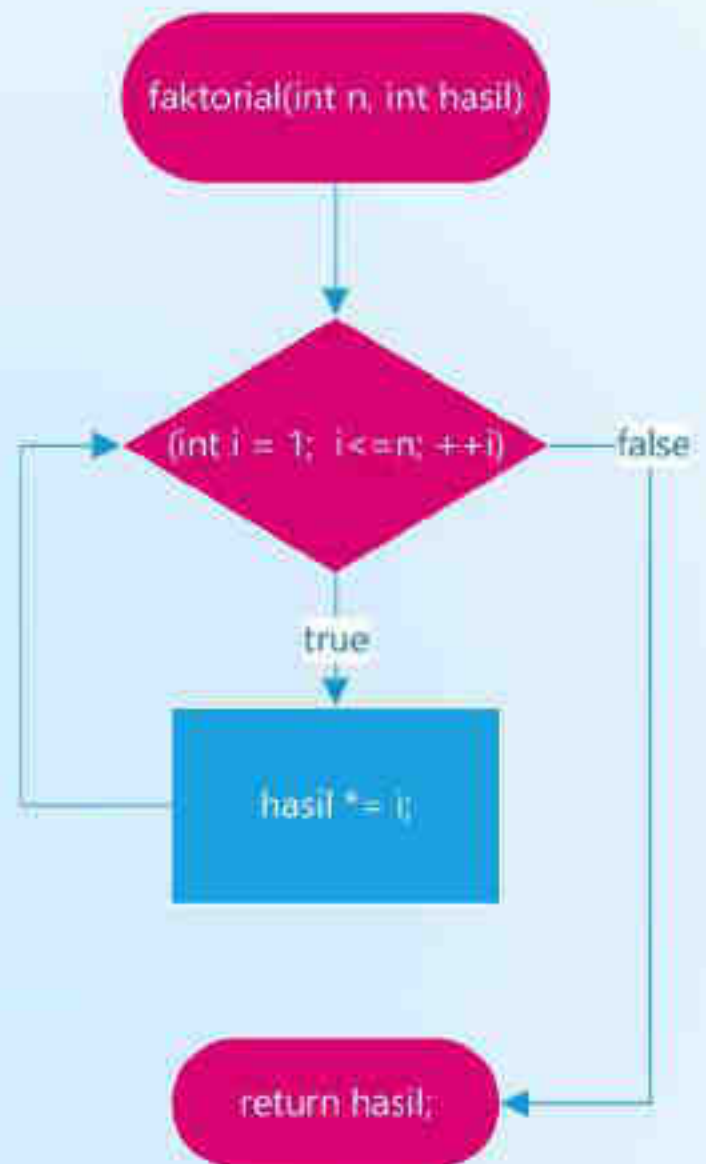
Output:

```
0: D:\Documents\De\Dev-C++\working\factorial.cpp
Masukkan bilangan bulat: 5
Faktorial dari 5! = 120
Process exited after 2.3431 seconds with return value 0
Press any key to continue . . .
```

Algoritma:

1. Buat variable n
2. Buat variable hasil diisi nilai 1
3. Masukkan bilangan bulat
4. Masukkan variable n dan hasil ke dalam fungsi factorial
5. Buat *loop* untuk mengalikan hasil dari perulangan sesuai nilai n
6. Kembalikan nilai hasil
7. Tampilkan hasil

Flowchart:



LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-7



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 7

```
genap_ganjil_fungsi.cpp
1  #include <iostream>
2  using namespace std;
3
4  int fungsiGenapGanjil(int angka)
5  {
6      if(angka % 2 == 0)
7      {
8          cout<<"Genap"<<endl;
9      }
10     else
11     {
12         cout<<"Ganjil"<<endl;
13     }
14 }
15
16 int main()
17 {
18     int angka;
19
20     cout<<"Masukkan angka: ";
21     cin>>angka;
22     fungsiGenapGanjil(angka);
23
24     return 0;
25 }
```

```
D:\Documents\file Dev C++\genap_ganjil_fungsi.exe
Masukkan angka: 15
Ganjil
.....
Process waited after 2.767 seconds with return value 0
Press any key to continue . . .
```

Source Code:

```
#include<iostream>
using namespace std;

int fungsiGenapGanjil(int angka)
{
    if(angka % 2 == 0)
    {
        cout<<"Genap"<<endl;
    }
    else
    {
        cout<<"Ganjil"<<endl;
    }
}

int main()
{
    int angka;

    cout<<"Masukkan angka: ";
    cin>>angka;
    fungsiGenapGanjil(angka);

    return 0;
}
```


Kesimpulan

Pada modul ke 7 ini saya bisa menarik kesimpulan, Saya memahami dasar-dasar fungsi (function). Sebuah fungsi dapat mengembalikan nilai ataupun tidak (void). Fungsi juga dapat menggunakan parameter maupun tidak.

Bentuk umum function:

tipeData namaFunction (parameter_1 (optional), parameter_n (optional))

```
{  
    // isi dari function  
    return namaVariabel; // optional  
}
```

Pemanggilan function:

namaFunction(isi_parameter_jika_ada);

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-8



Disusun Oleh:

Nama : Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama: Anzari Firman Saputra

Praktikum Algoritma

NIM: 20211402125

Laporan Awal - Pertemuan 2

Tema: Dasar

a) Pendahuluan

Array merupakan koleksi data dimana setiap elemen mempunyai nama dan tipe yg sama serta setiap elemen diakses dgn membacakan indeks array-nya.

Contoh: deklarasi:

```
int c [7];
```

inisialisasi nilai pada saat pendeklarasian:

```
int c [7] = {-45, 0, 6, 72, 1543, 43, 4};
```

b) Array dimensi satu

Bentuknya:

```
Type_data nama_var [ukuran];
```

c) Array dimensi dua

Bentuknya:

```
Type_data nama_var [ukuran] [ukuran];
```

Dan definisi variabel untuk setiap elemen tersebut adalah:

	0	1	2
0	b[0][0]	b[0][1]	b[0][2]
1	b[1][0]	b[1][1]	b[1][2]

d) Array Dimensi Banyak

Bentuknya:

```
Type_data nama_var [ukuran] [ukuran] [ukuran], dst;
```

Array ini seperti array dimensi dua tetapi dapat memiliki ukuran yg lebih besar.

Contoh:

```
static int datauser [2] [7] [8] [5];
```

TUGAS Penjabaran

1. Jelaskan yg dimaksud dgn Array!

Array merupakan koleksi data dimana setiap elemen memiliki nama dan tipe yg sama serta setiap elemen diakses dengan membedakan indeks array-nya.

2. Sebutkan dan jelaskan jenis-jenis array!

- array dimensi satu = array dgn satu indeks. `int var[5];`
- array dimensi dua = array dgn dua indeks. `int c[5][5];`
- array dimensi banyak = array dgn banyak indeks. `int var[2][9][9];`

3. Jelaskan kelebihan penggunaan array pada bahasa pemrograman C/C++!

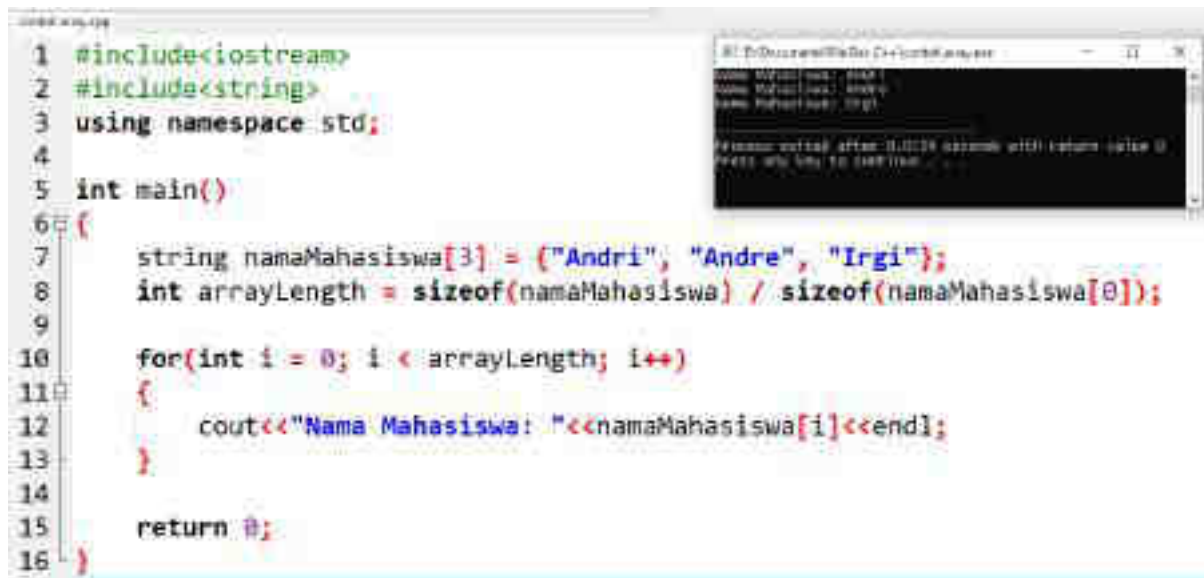
- Mudah digunakan untuk menyimpan data dgn tipe data dan nama variabel yg sama. Contoh: `String nama_mhs[100];`
- Mudah untuk menampilkan jumlah data yg banyak:

```
for(i=0; i<100; i++)  
{  
    cout << nama_mhs[i];  
    cout << endl;  
}
```
- Efisiensi yg tinggi
- Tidak berantakan

4. Buatlah contoh algoritma dan program sederhana menggunakan array dan memuat flowchart!

Jawaban No. 4

Contoh Program:



The screenshot shows a C++ program in a code editor on the left and its execution output in a console window on the right. The code defines an array of student names and prints each name. The console output shows the names 'Andri', 'Andre', and 'Irgi' printed on separate lines.

```
1 #include<iostream>
2 #include<string>
3 using namespace std;
4
5 int main()
6 {
7     string namaMahasiswa[3] = {"Andri", "Andre", "Irgi"};
8     int arrayLength = sizeof(namaMahasiswa) / sizeof(namaMahasiswa[0]);
9
10    for(int i = 0; i < arrayLength; i++)
11    {
12        cout<<"Nama Mahasiswa: "<<namaMahasiswa[i]<<endl;
13    }
14
15    return 0;
16 }
```

Console Output:

```
Andri
Andre
Irgi
Program exited after 0.0018 seconds with return value 0
Press any key to start the program...
```

Source Code:

```
#include<iostream>
#include<string>
using namespace std;

int main()
{
    string namaMahasiswa[3] = {"Andri", "Andre", "Irgi"};
    int arrayLength = sizeof(namaMahasiswa) /
sizeof(namaMahasiswa[0]);

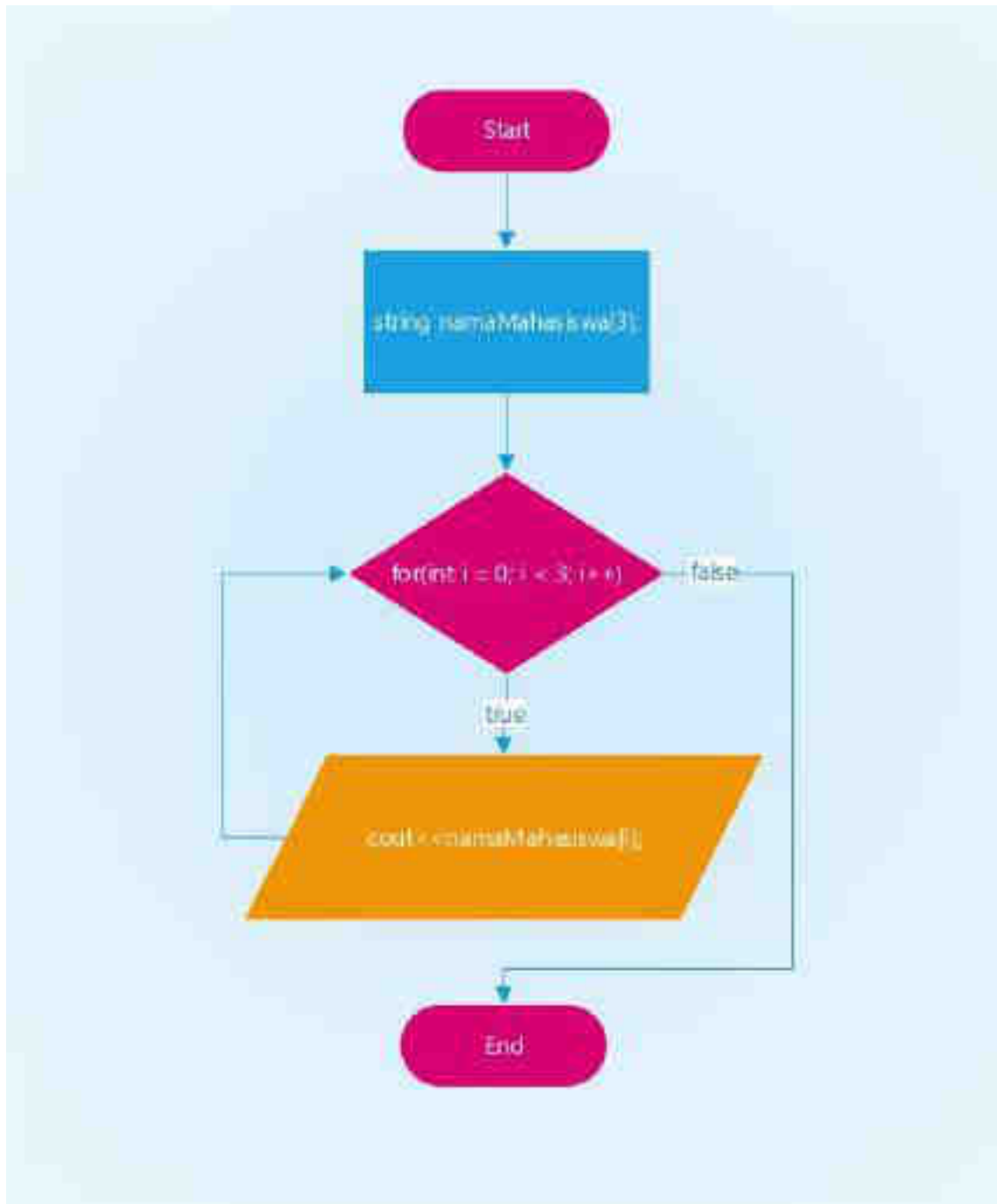
    for(int i = 0; i < arrayLength; i++)
    {
        cout<<"Nama Mahasiswa: "<<namaMahasiswa[i]<<endl;
    }

    return 0;
}
```

Algoritma:

1. Inisialisasi array namaMahasiswa dengan index 3 dan beri nilai 3 nama mahasiswa
2. Gunakan perulangan for untuk menampilkan nama mahasiswa
3. Nama mahasiswa telah ditampilkan

Flowchart:



LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-8



Disusun Oleh:

Nama : Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 8

The screenshot shows a C++ program in a code editor and its output in a terminal window.

Code Editor:

```

1 #include<iostream>
2 #include<iomanip>
3 using namespace std;
4
5 int main()
6 {
7     int a[3][1], b[3][1], c[3][1];
8     cout<<"Matrix A 3 x 1 = ";<<endl;
9
10    for(i=0;i<3;i++)
11    {
12        for(j=0;j<1;j++)
13        {
14            cout<<"Input Matrix ";<<(i+1)<<" = ";
15            cout<<"a["<<(i+1)<<"]<<"] = ";
16        }
17        cout<<endl;
18    }
19
20    cout<<"Matrix B 3 x 1 = ";<<endl;
21
22    for(i=0;i<3;i++)
23    {
24        for(j=0;j<1;j++)
25        {
26            cout<<"b["<<(i+1)<<"]<<"] = ";
27        }
28        cout<<endl;
29    }
30 }

```

Output Window:

```

Matrix A 3 x 1 =
Input Matrix 1 = a[0][0] = 1
Input Matrix 1 = a[1][0] = 2
Input Matrix 1 = a[2][0] = 3

Input Matrix 2 = a[0][1] = 1
Input Matrix 2 = a[1][1] = 2
Input Matrix 2 = a[2][1] = 3

Matrix B 3 x 1 =
b 1 1
b 2 2
b 3 3

Result: output after 0.01 seconds with return value 0
Press any key to exit.

```

Source Code:

```
#include<iostream>
#include<iomanip>
using namespace std;

int main()
{
    int a[3][3], b[3][3], c[3][3], i, j, k;

    cout<<"Matriks 3 x 3 : "<<endl;

    for(i=0;i<3;i++)
    {
        for(j=0;j<3;j++)
        {
            cout<<"Input Baris "<<(i+1)<<" , Kolom "<<(j+1)<<" = ";
            cin>>a[i][j];
        }
        cout<<endl;
    }

    cout<<"Matriks 3 x 3 = "<<endl<<endl;

    for(i=0;i<3;i++)
    {
        for(j=0;j<3;j++)
        {
            cout<<setw(4)<<a[i][j];
        }
        cout<<endl<<endl;
    }
}
```

Kesimpulan:

Pada modul ke 8 ini saya bisa menarik kesimpulan, Saya memahami dasar-dasar array. Array merupakan koleksi data di mana setiap elemen memakai nama dan tipe yang sama serta setiap elemen diakses dengan membedakan indeks array-nya.

Bentuk umum array 1 dimensi:

Tipe_data nama_variabel[ukuran];

Contoh: int jml_mhs[5];

Bentuk umum array 2 dimensi:

Tipe_data nama_variabel[ukuran][ukuran];

Contoh: char nama_nilai[5][3];

Bentuk umum array banyak dimensi:

Tipe_data nama_variabel[ukuran][ukuran][ukuran][ukuran][ukuran][ukuran] dst;

Contoh: int data_ktp[5][3][2][1][2][3][4][2];

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-9



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama: Andri Firman Saputra

NIM: 201011402125

Praktikum Algoritma
Laporan Awal - Pertemuan 9

Teori Dasar

a) Pendahuluan

String merupakan bentuk data yg bisa dipakai dalam pemrograman guna menyimpan dan memanipulasi data teks. Di dalam turbo c++, string bukan merupakan tipe data tersendiri tetapi hanya merupakan jenis khusus dari array.

String juga bisa diperlakukan dalam konstanta dan variabel.

b) Fungsi pada string

Pada Turbo C++ disediakan sejumlah fungsi macro yg didefinisikan di preprocessor `#define`. Macro-macro yg didefinisikan di `type.h` adalah:

1) `isalnum()`

Fungsi macro ini akan menghasilkan boolean true, kalau argumenta adalah huruf besar, huruf kecil dan karakter digit '0' sampai '9'.

Bentuk umum:

`type data isalnum (type argument);`

2) `isalpha()`

Fungsi macro ini akan menghasilkan boolean true, kalau argumenta adalah huruf luan angka.

Bentuk umum:

`type data isalpha (type argument);`

3) `isdigit()`

Fungsi macro ini akan menghasilkan boolean true, kalau argumenta adalah karakter digit '0' sampai '9'.

Bentuk umum:

`type data isdigit (type argument);`

4) `islower()`

Fungsi macro ini akan menghasilkan boolean true, kalau argumenta adalah huruf kecil.

Bentuk umum:

`type data islower (type argument);`

5) isupper()

Fungsi macro ini akan menghasilkan boolean true, kalau argumennya adalah huruf besar.

Bentuk umum:

type_data isupper (type argument);

6) tolower()

Fungsi Macro ini akan menghasilkan huruf kecil jika argumennya adalah huruf besar.

Bentuk umum:

type_data tolower (type argument);

7) toupper()

Fungsi macro ini kebalikan dari tolower, menghasilkan huruf besar jika argumennya adalah huruf kecil.

Bentuk umum:

type_data toupper (type argument);

8) strcpy()

Fungsi ini dipakai untuk menyalin string asal ke variabel string tujuan. Garis memiliki karakter '\0' yang menandakan seluruh karakter string asal.

Bentuk umum:

strcpy (tujuan, asal);

9) strlen()

Fungsi ini dipakai untuk menghitung jumlah karakter argumennya.

Bentuk umum:

strlen (var);

10) strcat()

Fungsi ini dipakai untuk menambahkan string asal ke akhir string tujuan.

Bentuk umum:

strcat (tujuan, asal);

11) strcmp()

Fungsi ini dipakai untuk membandingkan seri dengan seri.
Nilai hasil fungsi ini boolean true.
Jadi huruf 'A' lebih kecil dari 'B' itu.

12) strcmp()

fungsi ini dipakai untuk mengubah setiap huruf kapital ke huruf kecil di dalam string.

Contact:

```
char nama[3] = "Andi";
```

```
setlocale (LC_ALL);
```

WASTING: none

15) `strcmp()`

Fungsi ini akan mengolah semua huruf kecil pada string menjadi huruf kapital

$\subseteq B_n \in \mathcal{A}_n$.

```
char nama[7] = "Andri";
```

```
setvbuf (nama);
```

hasil = ANDA

TUGAS PENDAHULUAN

1. Jelaskan jenis-jenis fungsi string pada bahasa Pemrograman C/C++!

isminum(), isalpha(), isdigit(), islower(), isupper(), islower()

tolower(), strcpy(), strlen(), strcat(), strcmp(), strcmp(), strcmp()

2. Sebagai programmer, Mengapa anda menggunakan fungsi string?

Karena fungsi string mempunyai banyak kelebihan, seperti

- untuk memenuhi permintaan, produsen sering memindahkan

Seting, messabunjean seting.

3. Buatlah contoh program sederhana membuat beberapa variabel menggunakan aprat!

4. Buatlah contoh algoritma dan program sederhana menggunakan salah satu fungsi string untuk menghitung jumlah karakter dgn memakai flowchart!

3.

The image shows a C++ program in a code editor on the left and its execution output in a terminal window on the right. The code defines two character arrays, 'nama' and 'destinasi', both of size 25. 'nama' is initialized with the string "Andri Firman Saputra". The program then copies the content of 'nama' into 'destinasi' using 'strcpy'. Finally, it prints the contents of both arrays using 'cout'. The terminal window shows the output: 'Nama\t\t: Andri Firman Saputra' and 'Destinasi\t: Andri Firman Saputra', followed by a message indicating the program exited after 0.000 seconds with a return value of 0.

Source Code:

```
#include<iostream>
#include<string.h>
using namespace std;

int main()
{
    char nama[25] = {"Andri Firman Saputra"};
    char destinasi[25];

    strcpy(destinasi, nama);
    cout<<"Nama\t\t: "<<nama<<endl;
    cout<<"Destinasi\t: "<<destinasi;

    return 0;
}
```

4.



The screenshot shows a C++ program in a code editor on the left and its execution output in a terminal window on the right. The code defines a character array 'nama' with the string 'Andri Firman Saputra' and uses 'strlen' to calculate its length, which is then printed to the console. The terminal output shows the program name, the calculated length (17), and a message indicating the program ended successfully after 0.00135 seconds.

```
1 #include<iostream>
2 #include<string.h>
3 using namespace std;
4
5 int main()
6 {
7     char nama[25] = {"Andri Firman Saputra"};
8     cout<<"Panjang karakter: "<<strlen(nama);
9 }
```

Output from terminal:

```
Andri Firman Saputra
Panjang karakter: 17
Program ended after 0.00135 seconds with return value 0
Press any key to continue
```

Source Code:

```
#include<iostream>
#include<string.h>
using namespace std;

int main()
{
    char nama[25] = {"Andri Firman Saputra"};
    cout<<"Panjang karakter: "<<strlen(nama);
}
```

Algoritma:

1. Buat variabel bertipe char nama
2. Isi variabel dengan string
3. Panggil library #include<string.h>
4. Tampilkan fungsi strlen dengan parameter variabel nama

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-9



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 9

The image shows a screenshot of a C++ program in a code editor and its execution output in a terminal window. The code editor on the left shows a file named 'after pertemuan 9.cpp' with the following code:

```
1 #include<iostream>
2 #include<string>
3 using namespace std;
4
5 int main()
6 {
7     string nama, hobi;
8     cout<<"Masukkan Nama: ";
9     getline(cin, nama);
10    cout<<"Masukkan Hobi: ";
11    getline(cin, hobi);
12    cout<<endl;
13    cout<<"Jawaban Komputer"<<endl;
14    cout<<"Nama: "<<nama<<endl;
15    cout<<"Hobi: "<<hobi<<endl;
16 }
```

The terminal window on the right shows the output of the program. It displays the prompts 'Masukkan Nama: ' and 'Masukkan Hobi: ' followed by the user input 'Andri Firmans Saputra' and 'Mangruling'. The output then shows 'Jawaban Komputer' followed by 'Nama: Andri Firmans Saputra' and 'Hobi: Mangruling'. The terminal also shows a message 'Process exited after 2.415 seconds with return value 0' and 'Press any key to continue'.

Source Code:

```
#include<iostream>
#include<string>
using namespace std;

int main()
{
    string nama, hobi;
    cout<<"Masukkan Nama: ";
    getline(cin, nama);
    cout<<"Masukkan Hobi: ";
    getline(cin, hobi);
    cout<<endl;
    cout<<"Jawaban Komputer"<<endl;
    cout<<"Nama: "<<nama<<endl;
    cout<<"Hobi: "<<hobi<<endl;
}
```

Kesimpulan:

Pada modul ke 9 ini saya bisa membuat kesimpulan, Saya memahami tentang string lebih lanjut. String merupakan bentuk data yang biasa dipakai dalam pemrograman guna menampung dan memanipulasi teks. String juga bisa dipergunakan dalam konstanta dan variabel.

Fungsi – fungsi umum string yang sering digunakan:

1. isalnum()

Fungsi makro ini akan menghasilkan boolean kalau argumennya adalah huruf besar, huruf kecil dan karakter digit '0' sampai '9'.

2. isalpha()

Fungsi makro ini akan menghasilkan boolean kalau argumennya adalah huruf.

3. isdigit()

Fungsi makro ini akan menghasilkan boolean kalau argumennya adalah angka.

4. islower()

Fungsi makro ini akan menghasilkan boolean kalau argumennya adalah huruf kecil.

5. isupper()

Fungsi makro ini akan menghasilkan boolean kalau argumennya adalah huruf besar.

6. tolower()

Fungsi makro ini akan menghasilkan huruf kecil kalau argumennya adalah huruf besar.

7. toupper()

Fungsi makro ini akan menghasilkan huruf besar kalau argumennya adalah huruf kecil.

8. strcpy()

Fungsi ini dipakai untuk menyalin variabel string asal ke variabel string tujuan. variabel string tujuan harus memiliki ukuran lebih besar atau sama dengan string asal.

9. strlen()

Fungsi ini dipakai untuk menghitung jumlah karakter argumennya.

10. strcat()

Fungsi ini dipakai untuk menambahkan string asal ke bagian akhir string tujuan.
(menggabungkan string).

11. strcmp()

Fungsi ini dipakai untuk membandingkan dua buah string.

12. strlwr()

Fungsi ini dipakai untuk mengubah huruf besar ke huruf kecil pada string.

13. strupr()

Fungsi ini dipakai untuk mengubah huruf kecil ke huruf besar pada string.

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-10



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama: Andri Firman Saputra
NIM: 201011402125

Praktikum Algoritma
Laporan Awal - Pertemuan 10

Teori Dasar

a) Pendahuluan

Pointer adalah variabel yg berisi alamat memori sebagai penilaiannya dan berbeda dgn variabel biasa yg berisi nilai tertentu. Dengan kata lain, pointer berisi alamat dari variabel yg mempunyai nilai tertentu.

Bentuk Umum Pointer C++:

`type_data * variabel_name;`

* adalah operator memori yg fungsinya untuk menyesuaikan nilai variabel pada alamatnya yg ditentukan oleh operand.

Operator Pointer:

- 1) Operator alamat (&) adalah operator untuk get/set alamat memori yg dimiliki oleh variabel tersebut.
- 2) Operator dereferensi (*) adalah operator untuk get dari sebuah memori berdasarkan alamat memori.

Tugas Pendahuluan

1. Jelaskan yg dimaksud dgn Pointer!

Pointer adalah variabel yg berisi alamat memori sebagai penilaiannya dan berbeda dgn variabel biasa yg berisi nilai tertentu.

2. Sebutkan dan jelaskan jenis-jenis Pointer!

- Pointer aritmetika

Hanya 4 operator aritmetika yang digunakan pada pointer yaitu ++, --, + dan -.

- Pointer Array

Pointer dapat di-array seperti tipe data lainnya dalam C++

Contoh:

```
int *a[10];
```

3. Sebagai programmer, kenapa anda menggunakan pointer?
sangat memudahkan data dari satu fungsi ke fungsi lainnya
dan mengirim addressnya saja, tanpa mendatukannya.
4. Buatlah contoh algoritma dan program sederhana menggunakan
pointer dan memakai flowchart!

Jawaban No. 4



The screenshot shows a C++ IDE with two windows. The left window, titled 'peraturan.10 - gcrtest.cpp', contains the following source code:

```
1 #include<iostream>
2 using namespace std;
3
4 int main()
5 {
6     int x, *p, **q;
7
8     x = 10;
9     p = &x;
10    q = &p;
11    cout<<**q;
12
13    return 0;
14 }
```

The right window, titled 'A:\Select.D\Documents\Files\C++\peraturan.10 - gcrtest.exe', shows the program's execution output:

```
10
Process exited after 0.00186 seconds with return value 0
Press any key to continue . . .
```

Source Code:

```
#include<iostream>
using namespace std;

int main()
{
    int x, *p, **q;

    x = 10;
    p = &x;
    q = &p;
    cout<<**q;

    return 0;
}
```

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-10



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 10



Source Code:

```
#include<iostream>
```

```
using namespace std;
```

```
int HitungCharacter(char *s)
{
    int x = 0;
    for (x; *s != '\0'; s++)
    {
        ++x;
    }

    return x;
}
```

```
int main()
{
    char karakter[80];
    cout<<"Masukkan Kata: ";
    cin.getline(karakter,80);
    cout<<"Jumlah Karakternya Adalah "<<HitungCharacter(karakter)<<"
    Huruf"<<endl;
}
```

Kesimpulan:

Pada modul ke 10 ini saya bisa membuat kesimpulan, Saya memahami tentang pointer. Pointer adalah variabel yang berisi alamat memori sebagai penilaiannya dan berbeda dengan variabel biasanya yang berisi nilai tertentu. Dengan kata lain, pointer berisi alamat dari variabel yang mempunyai nilai tertentu.

Bentuk umum:

Tipe_data *variabel;

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-11



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Teori Dasar

a) Pendahuluan

Struktur (structure) adalah sekumpulan variabel yg masing-masing tipe datanya berbeda, dan dikelompokkan kedalam satu nama. (menurut Pascal struktur juga dikenal sebagai record). Struktur membantu data-data yg rumit, khususnya dalam program yg besar, karena struktur membiarkan sekelompok variabel diperlakukan sebagai satu unit.

Struktur ini sering digunakan untuk mendefinisikan suatu record data yang disimpan di dalam file. Struktur termasuk ke dalam tipe data yg dibangun (derived data type), yg disusun menggunakan objek dgn tipe yg lain.

Perhatikan definisi berikut:

```
struct mhs
```

```
{
```

```
    char *nama;
```

```
    char *nim;
```

```
    int uts, uas;
```

```
    float akhir;
```

```
    char mutu;
```

```
}
```

b) Struktur dan Fungsi

Operasi yg sering diterapkan pada struktur adalah proses mentain atau menunjukan struktur sebagai unit, menggunakan alamatnya dan mengakses anggotanya. copy and assignment mencakup memberi argument ke fungsi dan menghasilkan nilai dari fungsinya juga.

Struktur bisa diletakkan di awal oleh daftar variabel konstanta dan otomatis juga bisa ditempatkan di awal oleh operasi assignment. Sebuah struktur otomatis mungkin juga diletakkan di depan oleh tugas atau oleh panggilan fungsi yang menghasilkan struktur jenis yg tepat. Untuk mengklarifikasi nama struktur dan nama anggota digunakan simbol "."

c. Array dalam struktur

Array disini fungsinya sangat penting untuk menyimpan nama dan bilangan bulat yang akan digunakan dalam proses perhitungan.

d. Pointer dalam struktur

Misalkan sebuah pointer yaitu `ptPelajar`, yg menunjukan kepada sebuah data yg mempunyai struktur `PELAJAR` seperti berikut:

```
struct PELAJAR *ptPelajar;
```

Seperti pada pointer yg lain, deklarasi diatas tidak menyediakan sebarang tempat untuk record `PELAJAR`. Perlu dibuat record baru yg fungsinya menggunakan pointer, misalkan `Pelajar_baru`.

```
ptPelajar = &Pelajar_baru;
```

Dengan kondisi tersebut, pointer `ptPelajar` digunakan untuk mengganti tempat alamat `Pelajar_baru`, dan pointer `ptPelajar` ini ditunjukkan dgn menggunakan symbol \rightarrow seperti berikut ini:

```
ptPelajar  $\rightarrow$  nama = Khairul Ariffin;
```

```
ptPelajar  $\rightarrow$  kelas = 12345;
```

```
ptPelajar  $\rightarrow$  fakultas = Ekonomi;
```

Sama dengan :

```
* ptPelajar.nama = Khairudin;
```

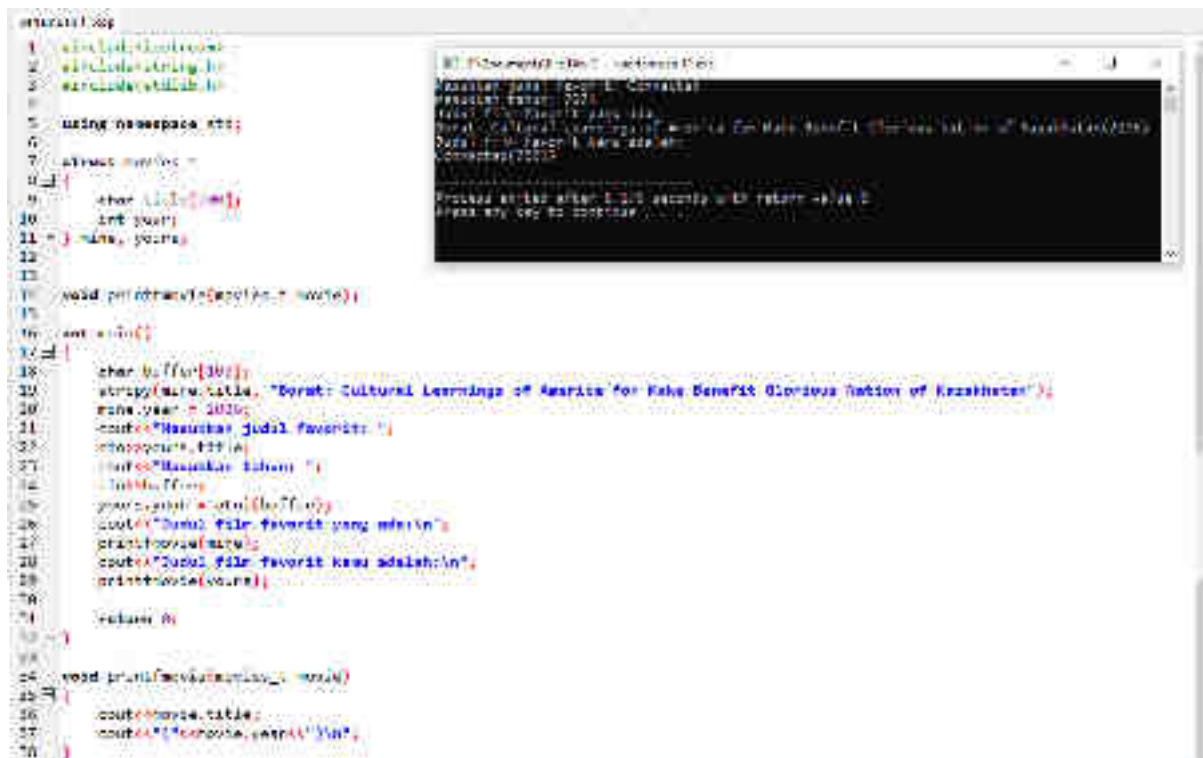
```
* ptPelajar.kelas = 12345;
```

```
* ptPelajar.fakultas = Ekonomi;
```


Tugas Pendahuluan

1. Jelaskan yg dimaksud dgn struktur pada bahasa C/C++!
struktur adalah sekumpulan variabel yg masing-masing tipe datanya berbeda, dan dikelompokkan ke dalam satu nama.
2. Sebutkan dan jelaskan jenis-jenis struktur!
 - struktur pemilihan: pemilihan langkah yg didasarkan oleh suatu kondisi atau pengambilan suatu keputusan.
 - struktur sekuensial: struktur dasar algoritma yg terdiri dari satu / lebih instruksi.
3. Sebagai programmer, mengapa anda menggunakan struktur?
struktur dapat membantu mengatur data-data yg rumit, khususnya dalam program yg besar, karena struktur membiarkan sekumpulan variabel diperlakukan sebagai satu unit daripada sebagai entry yg terpisah.
4. Buatlah contoh algoritma dan program sederhana menggunakan struktur dan memaui Flowchart!

Jawaban No. 4

The image shows a C++ program in a code editor and its execution output in a terminal window. The code defines a struct 'movies_t' with fields for title, year, and a union for favorite movies. It includes functions to print movie details and a main function that interacts with the user to input and display movie information. The terminal output shows the program's execution, including prompts for title, year, and favorite movies, and the resulting printed information.

```
1 #include<iostream>
2 #include<string.h>
3 #include<stdlib.h>
4
5 using namespace std;
6
7 struct movies_t
8 {
9     char title[100];
10    int year;
11    union {
12        char mine[100];
13        char yours[100];
14    };
15}
16
17 void printfmovie(movies_t movie)
18 {
19     cout<<endl;
20     char buffer[100];
21     strcpy(movie.title, "Dorab: Cultural Learnings of America for Make Benefit Glorious Nation of Kazakhstan");
22     movie.year = 2016;
23     cout<<"Masukkan judul favorit: ";
24     fgets(buffer, 100, stdin);
25     strcpy(movie.mine, buffer);
26     cout<<"Judul film favorit yang saya suka adalah: ";
27     printf("%s", movie.mine);
28     cout<<"Judul film favorit kamu adalah: ";
29     printf("%s", movie.yours);
30 }
31
32 int main()
33 {
34     void printfmovie(movies_t movie);
35     int year;
36     char title[100];
37     char buffer[100];
38     cout<<"Masukkan judul: ";
39     fgets(title, 100, stdin);
40     strcpy(title, buffer);
41     cout<<"Masukkan tahun: ";
42     fgets(buffer, 100, stdin);
43     year = atoi(buffer);
44     cout<<"Judul film favorit yang saya suka adalah: ";
45     printf("%s", title);
46     cout<<"Judul film favorit kamu adalah: ";
47     printf("%s", buffer);
48 }
```

```
13:51:10 [root@localhost ~]# ./main15.exe
Masukkan judul: Dorab: Cultural Learnings of America for Make Benefit Glorious Nation of Kazakhstan
Masukkan tahun: 2016
Masukkan judul favorit yang saya suka:
Masukkan judul favorit yang kamu suka:
Judul film favorit yang saya suka adalah: Dorab: Cultural Learnings of America for Make Benefit Glorious Nation of Kazakhstan
Judul film favorit kamu adalah:
Program ended after 0.115 seconds with return value 0
Press any key to continue . . .
```

Source Code:

```
#include<iostream>
```

```
#include<string.h>
```

```
#include<stdlib.h>
```

```
using namespace std;
```

```
struct movies_t
```

```
{
```

```
    char title[100];
```

```
    int year;
```

```
} mine, yours;
```

```
void printfmovie(movies_t movie);
```

```
int main()
```

```
{
```

```
    char buffer[100];
```

```

        strcpy(mine.title, "Borat: Cultural Learnings of America for Make Benefit
Glorious Nation of Kazakhstan");
        mine.year = 2006;
        cout<<"Masukkan judul favorit: ";
        cin>>yours.title;
        cout<<"Masukkan tahun: ";
        cin>>buffer;
        yours.year = atoi(buffer);
        cout<<"Judul film favorit yang ada:\n";
        printfmovie(mine);
        cout<<"Judul film favorit kamu adalah:\n";
        printfmovie(yours);

        return 0;
}

void printfmovie(movies_t movie)
{
    cout<<movie.title;
    cout<<"("<<movie.year<<")\n";
}

```

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-11



Disusun Oleh:

Nama: Andri Firman Saputra

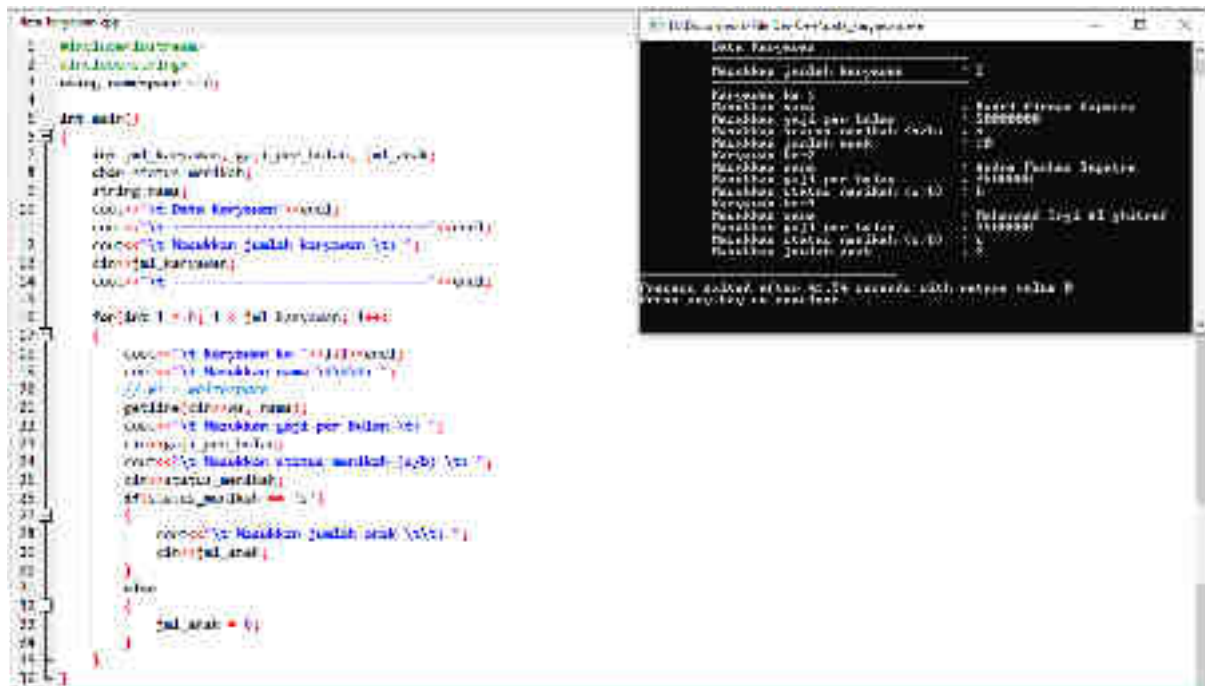
NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 11



Source Code:

```
#include<iostream>
#include<string>
using namespace std;

int main()
{
    int jml_karyawan, gaji_per_bulan, jml_anak;
    char status_menikah;
    string nama;

    cout<<"\t Data Karyawan"<<endl;
    cout<<"\t -----"<<endl;
    cout<<"\t Masukkan jumlah karyawan \t: ";
    cin>>jml_karyawan;
    cout<<"\t -----"<<endl;

    for(int i = 0; i < jml_karyawan; i++)
    {
        cout<<"\t Karyawan ke-"<<i+1<<endl;
        cout<<"\t Masukkan nama \t\t\t: ";
```



```

// ws = whitespace
getline(cin>>ws, nama);
cout<<"\t Masukkan gaji per bulan \t: ";
cin>>gaji_per_bulan;
cout<<"\t Masukkan status menikah (s/b) \t: ";
cin>>status_menikah;
if(status_menikah == 's')
{
    cout<<"\t Masukkan jumlah anak \t\t: ";
    cin>>jml_anak;
}
else
{
    jml_anak = 0;
}
}
}

```

Kesimpulan:

Pada pertemuan ke 11 ini saya dapat menarik kesimpulan, Saya memahami tentang struktur (structure). Struktur(structure) adalah sekumpulan variabel yang masing-masing tipe datanya berbeda, dan di kelompokkan ke dalam satu nama. (Menurut pascal struktur juga dikenal sebagai record). Struktur membantu data-data yang rumit, khususnya dalam program yang besar, Karena struktur membiarkan sekelompok variabel diperlakukan sebagai satu unit. Struktur ini sering digunakan untuk mendefinisikan suatu record data yang disimpan di dalam file. Struktur termasuk kedalam tipe data yang di bangkitkan (derived data type), yang disusun menggunakan objek dengan tipe yang lain.

Contoh bentuk umum struktur:

```
struct mhs
{
    char *nama;
    char *nim;
    int uts, uas;
    float akhir;
    char mutu;
}
```

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-12



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama : Andri Firman Saputra
NIM : 201011402125

Praktikum Algoritma
Laporan Awal - Pertemuan 12

Teori Dasar

a) Pendahuluan

Sorting adalah suatu proses pengurutan data yg sebelumnya disusun secara acak atau tidak teratur menjadi urut dan teratur menjadi urut dan teratur menurut suatu aturan tertentu.

Biasanya pengurutan terbagi menjadi 2 yaitu :

Ascending (pengurutan dari karakter / angka kecil ke karakter / angka besar ke karakter / angka kecil)

Descending kebalikannya.

1) Bubble sort

Bubble sort adalah suatu metode pengurutan yg membandingkan elemen yg sekarang dgn elemen berikutnya. Perbandingan aramahnya dapat dimulai dari data yg paling awal atau yg paling akhir.

Apabila elemen yg sekarang (sebelumnya) lebih besar dari elemen berikutnya, maka posisi ditukar, kalau tidak posisinya tetap atau tidak perlu ditukar.

2) Selection Sort

Selection sort adalah suatu metode pengurutan yg membandingkan elemen yg sekarang dgn elemen berikutnya sampai elemen terakhir.

Jika ditemukan elemen lain yg lebih kecil dari elemen sekarang maka dicatat posisinya dan langsung ditukar.

3) Quick Sort

Quick sort adalah suatu metode pengurutan yg membandingkan suatu elemen (pivot) dengan elemen yg lain dan menyusunnya sedemikian rupa sehingga elemen

yg lain lebih kecil dari pada pivot terletak disebelah kiri pivot sedangkan elemen yg lebih besar dari pivot diletakkan disebelah kanan pivot.

4) Merge Sort

Merge Sort adalah suatu metode pengurutan yg membandingkan elemen. Satu dgn elemen yg lain, apabila nilai nya lebih kecil maka datanya ditampung di elemen yg lain lagi.

Tugas Pendahuluan

1. Apa yg dimaksud dgn Sorting?
Sorting adalah suatu proses pengurutan data yg sebelumnya disusun secara acak atau tidak teratur menjadi urut dan teratur menurut suatu aturan tertentu.
2. Sebutkan dan jelaskan jenis-jenis Sorting pada bahasa C/C++!
 - 1) Bubble Sort
 - 2) Selection Sort
 - 3) Quick Sort
 - 4) Merge Sort
3. Sebagai Programmer, mengapa anda menggunakan Sorting?
Karena jika tidak menggunakan sorting data akan menjadi berantakan dan sulit di baca, sebaliknya jika kita menggunakan sorting data lebih mudah dibaca dan diolah.
4. Buatlah contoh algoritma dan program sederhana menggunakan sorting dgn memakai flowchart!

Contoh Program Sorting:

```
1 //BubbleSort.cpp
2 using namespace std;
3
4 void BubbleSort(int data[], int n)
5 {
6     int i, j, temp;
7     for(i = 0; i < n-1; i++)
8     {
9         for(j = 0; j < n-i-1; j++)
10         {
11             if(data[j] > data[j+1])
12             {
13                 temp = data[j];
14                 data[j] = data[j+1];
15                 data[j+1] = temp;
16             }
17         }
18     }
19 }
20
21 int main()
22 {
23     int i, j, temp;
24     int data[] = {10, 40, 30, 50, 20, 45, 15, 25, 100, 200, 300};
25     cout << "Pengurutan Bubblesort:" << endl;
26     cout << "-----" << endl;
27     cout << "Data sebelum diurutkan: ";
28     for(i = 0; i < n; i++)
29     {
30         cout << data[i] << " ";
31     }
32     cout << endl;
33     BubbleSort(data, n);
34     cout << "Data setelah diurutkan: ";
35     for(i = 0; i < n; i++)
36     {
37         cout << data[i] << " ";
38     }
39     cout << endl;
40     return 0;
41 }
42
43 void SelectionSort(int array[], int n)
44 {
45     int i, j, temp;
46     for(i = 0; i < n-1; i++)
47     {
48         int min = i;
49         for(j = i+1; j < n; j++)
50         {
51             if(array[j] < array[min])
52             {
53                 min = j;
54             }
55         }
56         temp = array[i];
57         array[i] = array[min];
58         array[min] = temp;
59     }
60 }
```

```
1 //BubbleSort.cpp
2
3 Data sebelum diurutkan:
4 10 40 30 50 20 45 15 25 100 200 300
5
6 Data setelah diurutkan:
7 10 15 20 25 30 40 45 50 100 200 300
8
9 Process exited with 0 bytes memory with error code 0
10 Press any key to continue . . .
```

Source Code:

```
#include<iostream>
using namespace std;
void BubbleSort(int data[], int n);

int main()
{
    int i, n = 9, data[] = {10, 40, 20, 60, 15, 4, 8, 100, 200, 25};

    cout<<"Pengurutan Menaik"<<endl;
    cout<<"-----"<<endl;
    cout<<"Sebelum diurutkan: "<<endl;

    for(i = 0; i <= n; i++)
    {
        cout<<data[i]<<" ";
    }

    cout<<endl;

    cout<<"-----"<<endl;

    BubbleSort(data, n);

    cout<<"Setelah diurutkan: "<<endl;

    for(i = 0; i <= n; i++)
    {
        cout<<data[i]<<" ";
    }

    cout<<endl;

    return 0;
}
```



```
void BubbleSort(int array1[], int n)
{
    int i, j, tmp;

    for(i = 1; i <= n; i++)
    {
        for(j = n; j >= i; j--)
        {
            // controller asc and desc
            if(array1[j] < array1[j-1])
            {
                tmp = array1[j];

                array1[j] = array1[j-1];

                array1[j-1] = tmp;
            }
        }
    }
}
```

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-12



Disusun Oleh:

Nama: Andri Firman Saputra

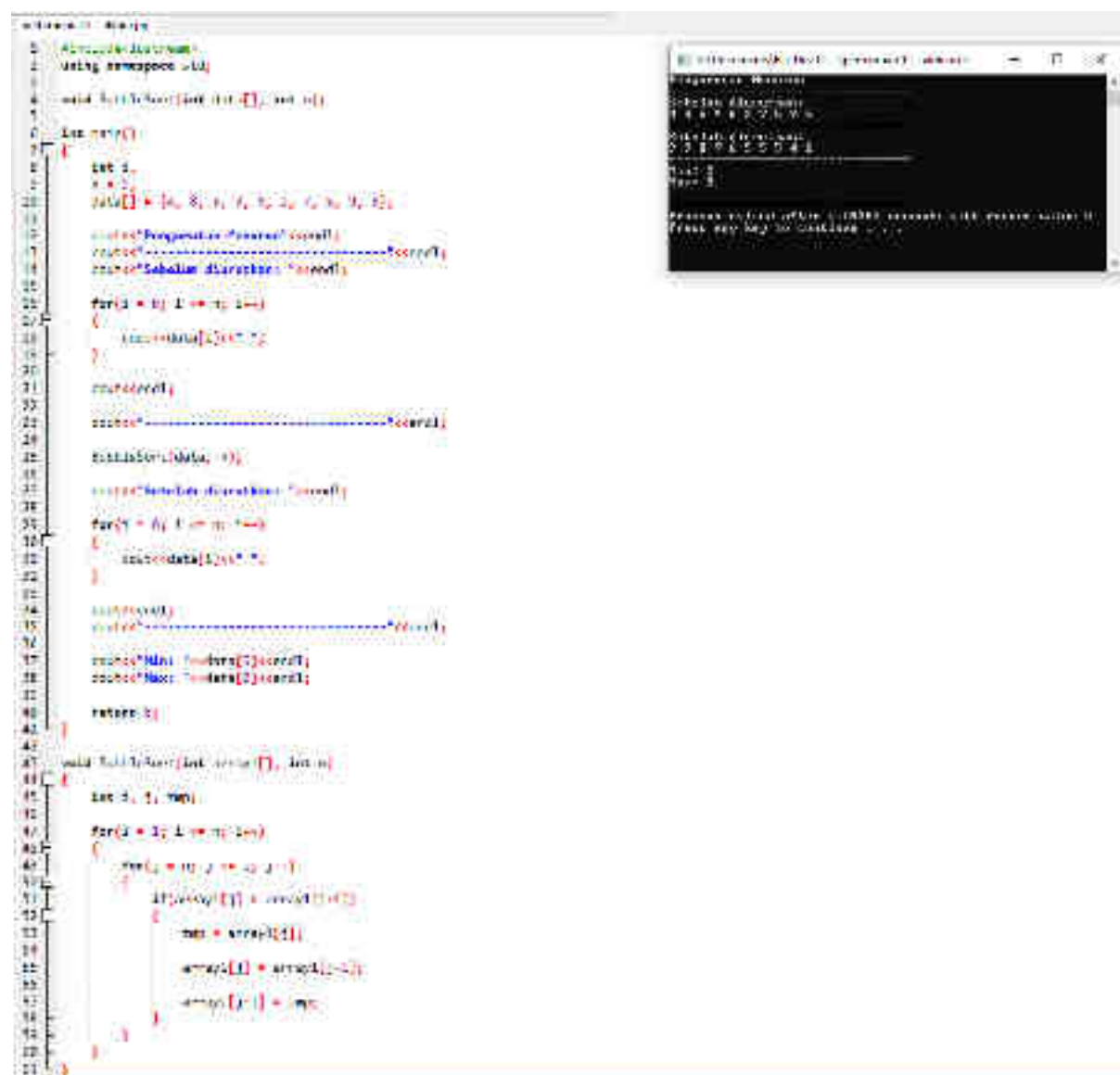
NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 12



Source Code:

```
#include<iostream>

using namespace std;

void BubbleSort(int data[], int n);

int main()
{
    int i,
    n = 9,
    data[] = {4, 8, 5, 9, 6, 2, 7, 5, 9, 5};

    cout<<"Pengurutan Menurun"<<endl;
    cout<<"-----"<<endl;
    cout<<"Sebelum diurutkan: "<<endl;

    for(i = 0; i <= n; i++)
    {
        cout<<data[i]<<" ";
    }

    cout<<endl;

    cout<<"-----"<<endl;

    BubbleSort(data, n);

    cout<<"Setelah diurutkan: "<<endl;

    for(i = 0; i <= n; i++)
    {
        cout<<data[i]<<" ";
    }

    cout<<endl;
```

```

        cout<<"-----"<<endl;

        cout<<"Min: "<<data[9]<<endl;
        cout<<"Max: "<<data[0]<<endl;

        return 0;
    }

void BubbleSort(int array1[], int n)
{
    int i, j, tmp;

    for(i = 1; i <= n; i++)
    {
        for(j = n; j >= i; j--)
        {
            if(array1[j] > array1[j-1])
            {
                tmp = array1[j];

                array1[j] = array1[j-1];

                array1[j-1] = tmp;
            }
        }
    }
}

```

Kesimpulan:

Pada pertemuan ke 12 ini saya dapat menarik kesimpulan, Sorting adalah suatu proses pengurutan data yang sebelumnya disusun secara acak atau tidak teratur menjadi urut dan teratur menjadi urut dan teratur menurut suatu aturan tertentu. Biasanya pengurutan terbagi menjadi 2 yaitu : ascending (pengurutan dari karakter/ angka kecil ke karakter / angka besar ke karakter/ angka kecil) Ada banyak cara yang dapat dilakukan untuk melakukan proses pengurutan dari paling atas ke paling bawah atau sebaliknya. Untuk melakukan proses pengurutan dapat menggunakan beberapa metode antara lain:

1) Bubble sort

Bubble sort adalah suatu metode pengurutan yang membandingkan elemen yang sekarang dengan elemen berikutnya. Perbandingan alamatnya dapat dimulai dari data yang paling awal atau yang paling akhir. Apabila elemen yang sekarang (sebelumnya) lebih besar dari elemen berikutnya, maka posisi di tukar, kalau tidak posisinya tetap atau tidak perlu ditukar.

2) Selection sort

Selection sort adalah suatu metode pengurutan yang membandingkan elemen yang sekarang dengan elemen berikutnya sampai elemen terakhir. Jika ditemukan elemen lain yang lebih kecil dari elemen sekarang maka di catat posisinya dan langsung ditukar.

3) Quick sort

Quick sort adalah suatu metode pengurutan yang membandingkan suatu elemen (pivot) dengan elemen yang lain dan menyusun sedemikian rupa sehingga elemen yang lain lebih kecil dari pada pivot terletak disebelah kiri pivot sedangkan elemen yang lebih besar dari pivot diletakan disebelah kanan pivot.

4) Merge sort

Merge sort adalah suatu metode pengurutan yang membandingkan elemen satu dengan elemen yang lain, apabila nilainya lebih kecil maka datanya ditampung di elemen yang lain lagi.

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-13



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama : Andri Firman Saputra
NIM : 201011402125

Praktikum Algoritma
Labirin Awal - Peremuan 13

Teori Dasar

a) Pendahuluan

Proses pencarian suatu elemen di dalam array disebut searching, ada 2 macam pencarian yaitu pencarian sekuensial (sequential searching) dan pencarian biner (binary searching). Perbedaannya terletak pada keadaan suatu elemen atau data yg berada pada array. Pencarian sekuensial digunakan apabila data dalam keadaan acak atau tidak urut. Sedangkan pencarian biner digunakan pada data yg sudah dalam keadaan urut.

b) Pencarian Sekuensial

Pencarian sekuensial menggunakan prinsip sebagai berikut: data yg ada pada suatu array dibandingkan satu persatu dgn data yg dicari.

Pencarian ini dilakukan dgn melakukan suatu pengulangan dari 1 sampai semua data yg ada. Pada setiap kali pengulangan, dibandingkan data yg posisinya ke - i dengan data yg dicari atau dimasukkan. Apabila sama, maka data tersebut telah ditemukan dan proses pengulangan dihentikan.

Sebaliknya, kalau sampai pengulangan selesai dan data yg dicari tidak ditemukan, maka tsb tidak ada.

Tugas Pendahuluan

1. Apa yg dimaksud dgn searching?

Searching adalah proses pencarian suatu elemen di dm array.

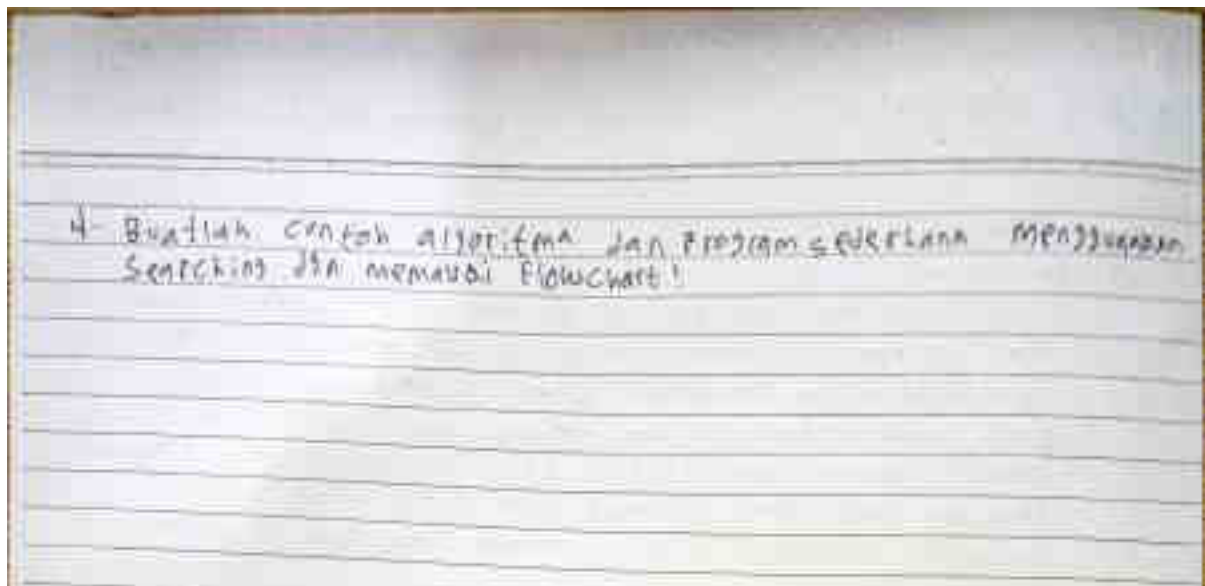
2. Jelaskan kelebihan dan kekurangan Searching pada bahasa C/C++!

Kelebihan: apabila data yg dicari letaknya pada data-dan awal sehingga prosesnya berjalan cepat.

Kekurangan: apabila data yg dicari letaknya pada data terakhir maka dalam penggunaan waktu, proses ini berjalan lama.

3. Sebagai programmer, mengapa anda menggunakan searching? karena dapat memudahkan pencarian data.





Jawaban No. 4

```

1 //Selection Sort Program
2 using namespace std;
3
4 void Hirarchy(int data[], int n, int a, int kida);
5
6 int main()
7 {
8     int data[] = {1, 30, 25, 10, 45, 75, 60, 70, 80, 90};
9     int kida, n, i, j;
10     j = 0;
11
12     cout << "Elemen Array : ";
13     for(i = 0; i < j; i++)
14     {
15         cout << data[i] << " ";
16     }
17     cout << endl;
18
19     cout << "Masukkan data yang akan dicari : ";
20     cin >> kida;
21
22     Hirarchy(data, j, kida, a, kida);
23     if(i % 10 == 0)
24     {
25         cout << "Data yang dicari berada pada list ke : " << i << endl;
26     }
27     else
28     {
29         cout << "Data yang dicari tidak ada dalam array" << endl;
30     }
31 }
32
33 void Hirarchy(int data[], int n, int a, int kida)
34 {
35     bool ketemu = false;
36     int top = 0, bottom = 0, mid;
37     int i = 0;
38     while(bottom <= top && !ketemu)
39     {
40         mid = (top + bottom) / 2;
41
42         if(data[mid] == kida)
43         {
44             ketemu = true;
45         }
46         else
47         {
48             if(data[mid] > kida)
49             {
50                 top = mid - 1;
51             }
52             else
53             {
54                 bottom = mid + 1;
55             }
56         }
57     }
58
59     if(ketemu)
60     {
61         cout << "Data : " << kida << endl;
62     }
63     else
64     {
65         cout << "Data : " << kida << endl;
66     }
67 }

```

Source Code:

```
#include<iostream>

using namespace std;

void BinSearch(int data[], int n, int x, int *idx);

int main()
{
    int data[] = {1, 16, 25, 30, 45, 55, 68, 75, 82, 93};
    int idx, x, i, jmlData = 10;

    cout<<"Elemen Array: ";
    for(i = 0; i < jmlData; i++)
    {
        cout<<data[i]<<" ";
    }
    cout<<endl;

    cout<<"Masukkan data yang akan dicari: ";
    cin>>x;

    BinSearch(data, jmlData, x, &idx);
    if(idx != -1)
    {
        cout<<"Data yang dicari berada pada indeks: "<<idx<<endl;
    }
    else
    {
        cout<<"Data yang dicari tidak ada dalam array"<<endl;
    }
}

void BinSearch(int data[], int n, int x, int *idx)
{
```

```

bool ketemu = false;
int top = n-1, bottom = 0, mid;
int i = 0;
while(bottom <= top && !ketemu)
{
    mid = (top + bottom) / 2;

    if(data[mid] == x)
    {
        ketemu = true;
    }
    else
    {
        if(data[mid] > x)
        {
            top = mid - 1;
        }
        else
        {
            bottom = mid + 1;
        }
    }
}

if(ketemu)
{
    *idx = mid;
}
else
{
    *idx = -1;
}
}

```

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-13



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 13

```

1 // Fungsi untuk mencari data dalam array
2 using namespace std;
3
4 int linearSearch(int data[], int dataSize, int element);
5
6 int main()
7 {
8     int dataSize = 10;
9     int data[dataSize];
10    int searchingElement = 0;
11    int flag = 0;
12    cout << "Masukkan 10 buah data berikut: ";
13    for(int i = 0; i < dataSize; i++)
14    {
15        cout << "Masukkan data ke-" << (i+1) << ": ";
16        data[i] = 0;
17    }
18
19    cout << "\n\nMau array apakah? ";
20    cout << "array 1: 10, 20, 30, 40, 50, 60, 70, 80, 90, 100";
21    cout << "array 2: 10, 20, 30, 40, 50, 60, 70, 80, 90, 100";
22    int array = 0;
23    while(array < 1 || array > 2)
24    {
25        cout << "Masukkan array yang akan dicari: ";
26        array = 0;
27    }
28
29    cout << "Masukkan data yang dicari: ";
30    searchingElement = 0;
31
32    flag = linearSearch(data, dataSize, searchingElement);
33
34    if(flag == -1)
35    {
36        cout << "Data tersebut ditemukan pada posisi array: " << flag << endl;
37    }
38    else
39    {
40        cout << "Data tersebut tidak ditemukan" << endl;
41    }
42
43    return 0;
44 }
45
46 int linearSearch(int data[], int dataSize, int element)
47 {
48     int flag = -1;
49     for(int col = 0; col < dataSize; col++)
50     {
51         if(data[col] == element)
52         {
53             flag = col;
54             break;
55         }
56     }
57
58     return flag;
59 }

```

Source Code:

```
#include<iostream>

using namespace std;

int linearSearch(int data[], int dataSize, int element);

int main()
{
    int dataSize = 5;
    int data[dataSize];
    int searchingElement = 0;
    int flag = 0;
    cout<<"Masukkan 5 buah data berikut: "<<endl;
    for(int i = 0; i < dataSize; i++)
    {
        cout<<"Masukkan data ke "<<i+1<<": ";
        cin>>data[i];
    }

    cout<<"Isi dari array adalah: "<<endl;
    cout<<"Array: "<<"\t\t\t"<<"data: "<<endl;
    for(int count = 0; count < dataSize; count++)
    {
        cout<<"array["<<count<<"]"<<"\t\t";
        cout<<data[count]<<endl;
    }

    cout<<"\n\nMasukkan data yang dicari: ";
    cin>>searchingElement;

    flag = linearSearch(data, dataSize, searchingElement);
```



```

        if(flag != -1)
        {
            cout<<"\n\nData tersebut ditemukan pada posisi: array["<<flag<<"]";
        }
        else
        {
            cout<<"\n\nData tersebut tidak ditemukan"<<endl;
        }

        return 0;
    }

int linearSearch(int data[], int dataSize, int element)
{
    int flag = -1;
    for(int count = 0; count < dataSize; count++)
    {
        if(element == data[count])
        {
            flag = count;
            break;
        }
    }
    return flag;
}

```

Kesimpulan:

Pada pertemuan ke 13 ini saya dapat menarik kesimpulan mengenai searching yaitu, searching adalah proses pencarian suatu elemen di dalam array, ada 2 macam pencarian yaitu pencarian sekuensial (sequential searching) dan pencarian biner (binary searching). Perbedaannya terletak pada keadaan suatu elemen atau data yang berada pada array. Pencarian sekuensial digunakan apabila data dalam keadaan acak atau tidak urut. Sedangkan pencarian biner digunakan pada data yang sudah dalam keadaan urut.

Pencarian Sekuensial

Pencarian sekuensial menggunakan prinsip sebagai berikut: data yang ada pada suatu array dibandingkan satu persatu dengan data yang dicari. Pencarian ini dilakukan dengan melakukan suatu pengulangan dari 1 sampai semua data yang ada. Pada setiap kali pengulangan, dibandingkan data yang posisinya ke-i dengan data yang dicari atau maksud. Apabila sama, maka data tersebut telah ditemukan dan proses pengulangan dihentikan. Sebaliknya, kalau sampai pengulangan selesai dan data yang dicari tidak ditemukan, maka tersebut tidak ada.

Pencarian Biner

Pencarian biner adalah sebuah teknik untuk menemukan nilai tertentu dalam sebuah larik (array), dengan menghilangkan setengah data pada setiap langkah, dipakai secara luas tetapi tidak secara eksklusif dalam ilmu komputer. Sebuah pencarian biner mencari nilai tengah (median), melakukan sebuah perbandingan untuk menentukan apakah nilai yang dicari ada sebelum atau sesudahnya, kemudian mencari setengah sisanya dengan cara yang sama. Pencarian biner digunakan pada data yang sudah dalam keadaan urut.