

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

Ringkasan Materi

a) Pendahuluan

Bahasa C++ merupakan bahasa pemrograman tingkat tinggi, yaitu 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 se kali saja dalam satu program.

b.) Tipe Data

1) Tipe Data Integer

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

Juga juga tipe data integer yg dideklarasikan tidak bertanda (unsigned integer) dg menggunakan kata kunci unsigned.

2) Tipe Data Pecahan (Float)

Tipe data pecahan atau floating point dapat dideklarasikan dg 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 melakukn 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 dielompokkan sebagai berikut :

1) Operator Pengisian (Assignment Operator)

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

Contoh:

$$a = 5$$

$$c = b = a$$

maka nilai a dan b sama dengan c yaitu 5.

2) Operator aritmatika (arithmetic Operator)

- Operator aritmatika yg tergolong sebagai operator binary.

| | |
|---|---------------------------|
| * | Perkalian |
| / | Pembagian |
| % | Modulus atau Sisa pembagi |
| + | 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 dg 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 memisahkan beberapa ungkapan yg dipisahkan dgn tanda koma di dalam kurung buka dan kurung tutup.

9). Operator gabungan

C++ memungkinkan penulisan suatu operator pengajaran (=) Untuk digabung dgn operator lainnya yaitu dgn operator aritmatika, operator peningkatan dan penurunan serta operator bitwise.

Bentuk penulisannya sebagai berikut:

operator Pengajaran

Variabel = Variabel operator Ungkapan

Dapat diulah dengan tulisan sebagai berikut:

operator lainnya

operator Pengajaran

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 pemangkatan

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



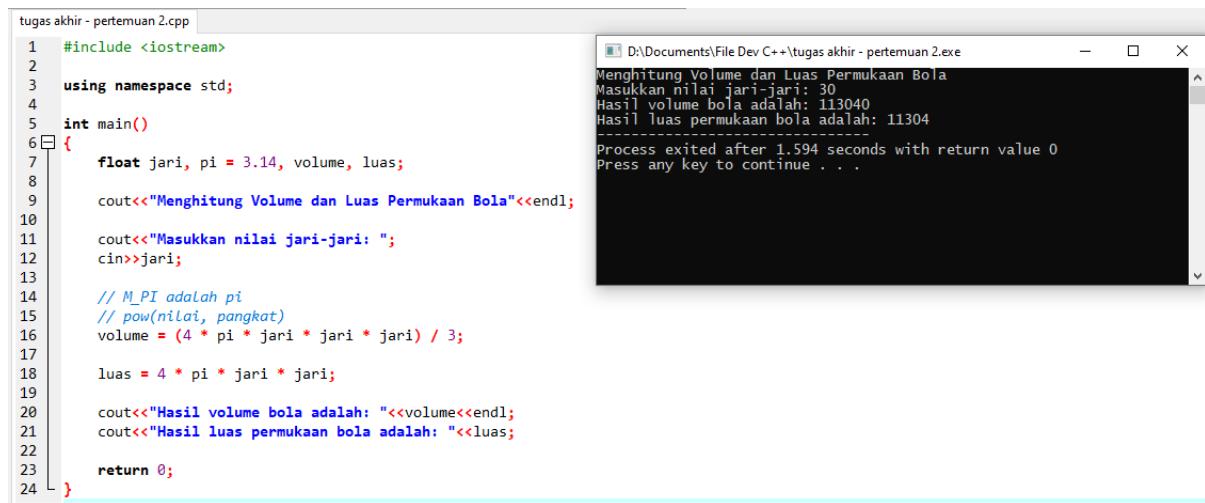
Disusun Oleh:

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 screenshot shows a code editor window titled "tugas akhir - pertemuan 2.cpp" containing C++ code for calculating the volume and surface area of a sphere. The code includes prompts for input, calculations using the formula $V = \frac{4}{3}\pi r^3$ and $A = 4\pi r^2$, and output statements for the results. To the right, a terminal window titled "D:\Documents\File Dev C++\tugas akhir - pertemuan 2.exe" shows the program's execution. It asks for the radius, calculates the volume as 113040, and the surface area as 11304. The process exits after 1.594 seconds.

```
1 #include <iostream>
2
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    cin>>jari;
13
14 // M_PI adalah pi
15 // pow(nilai, pangkat)
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;
22
23 return 0;
24 }
```

```
D:\Documents\File Dev C++\tugas akhir - pertemuan 2.exe
Menghitung Volume dan Luas Permukaan Bola
Masukkan nilai jari-jari: 30
Hasil volume bola adalah: 113040
Hasil luas permukaan bola adalah: 11304
Process exited after 1.594 seconds with return 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 Saputra

Pratinum Algoritma

NIM : 201011402125

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 bekend dan end.

Perintah IF memiliki bentuk umum:

if (kondisi) {
pernyataan}

}

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

bisa juga seperti ini :

if (kondisi)
{
pernyataan 1
pernyataan 2
dst}

}

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 majemuk 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 atau pun pernyataan majemuk.

If dalam IF : pernyataan IF yg terletak di dalam IF.

Atau bisa disebut nested-if

3. mengapa anda sebagai programmer harus menggunakan perintah dalam program?

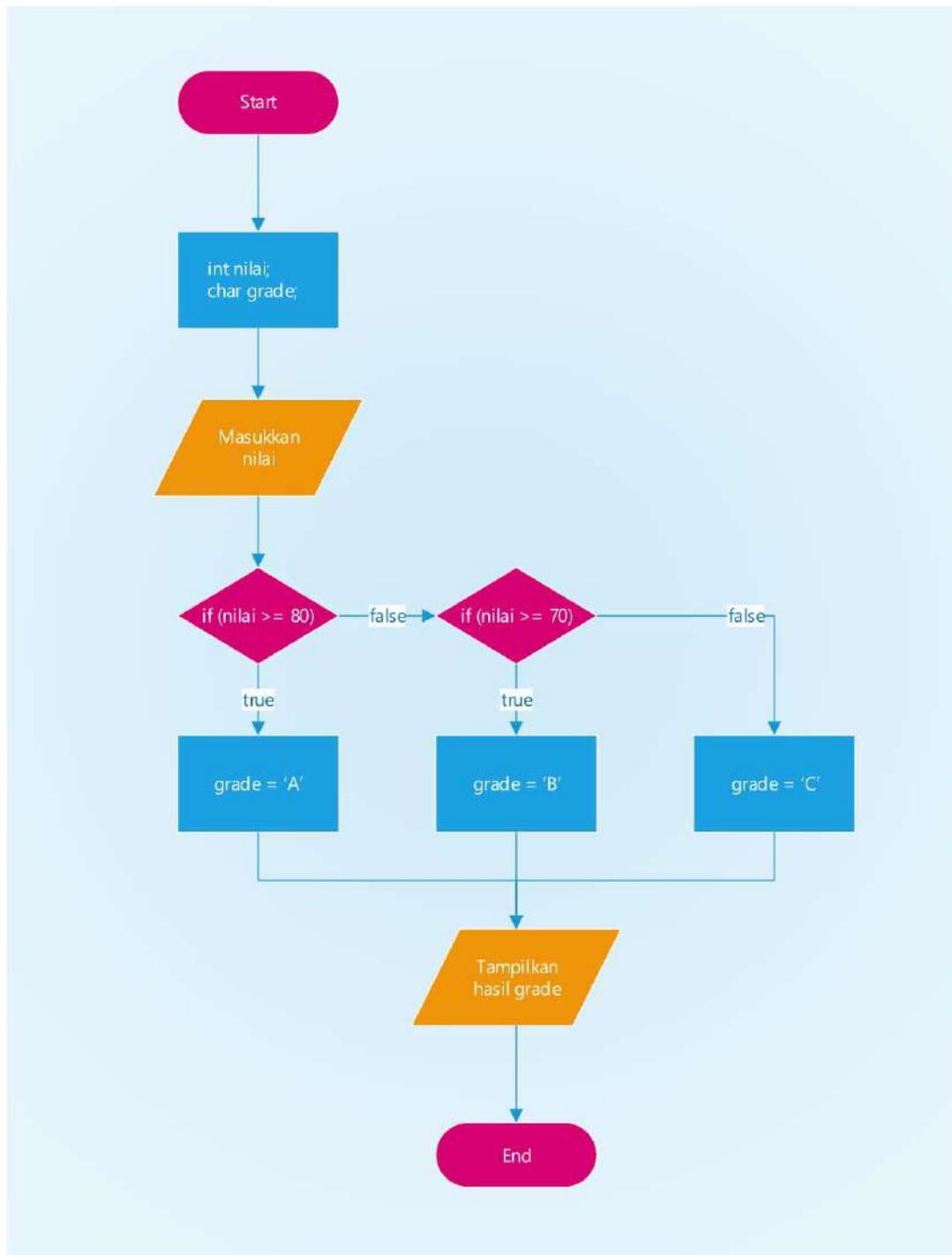
Karena, semua program membutuhkan pernyataan,
seperti keluar aplikasi, menghapus data atau bukaan
algoritma program.

4. Buatlah algoritma dan program sederhana menggunakan perintah if (gn memauai Flowchart).

Algoritma:

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

Flowchart



Program:

The screenshot shows a C++ development environment with two windows. On the left is the code editor window titled "laporan awal - pertemuan 3.cpp", containing the following C++ code:

```
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
29    return 0;
30 }
```

On the right is the terminal window titled "D:\Documents\File Dev C++\laporan awal - pertemuan 3.exe", showing the program's output:

```
D:\Documents\File Dev C++\laporan awal - pertemuan 3.exe
Program sederhana menggunakan IF
Masukkan Nilai: 95
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 screenshot shows a code editor window titled "pertemuan 3 laporan akhir tugas akhir.cpp" and a terminal window. The code in the editor is a C++ program that prompts the user to enter a day code (1-7) and then prints the corresponding day name. The terminal window shows the program's output after it has been run.

```
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 }
```

D:\Documents\File Dev C++\pertemuan 3 laporan akhir tug... — X
Masukkan kode: 1
Senin

Process exited after 1.13 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

Praktium Algoritma

NIM : 201011402125

Laporan Awal - Pertemuan 4

Teori Dasar

a) Pendahuluan

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

Bentuk Umum SWITCH

switch (Ekspresi)

{

CASE Konstanta1 :

Pernyataan1 ;

break;

CASE Konstanta2 :

Pernyataan2 ;

break;

default :

Pernyataan N;

}

Ekspresi bisa berupa integer atau char.

Setiap Konstanta1, Konstanta2, dst. dapat berupa konstanta integer atau konstanta char.

Setiap case dapat menampung satu atau lebih pernyataan.

Pengujian pada switch dimulai dari case yg paling atas.

Jika nilai pada case cocok dg ekspresinya maka akan dijalankan case tersebut. Tapi jika tidak cocok maka akan beralih ke case berikutnya sampai default jika ada.

Perintah break berfungsi untuk mengakhiri ekspresi. Jika tidak ada break case berikutnya diberikan.

TUGAS Pendahuluan

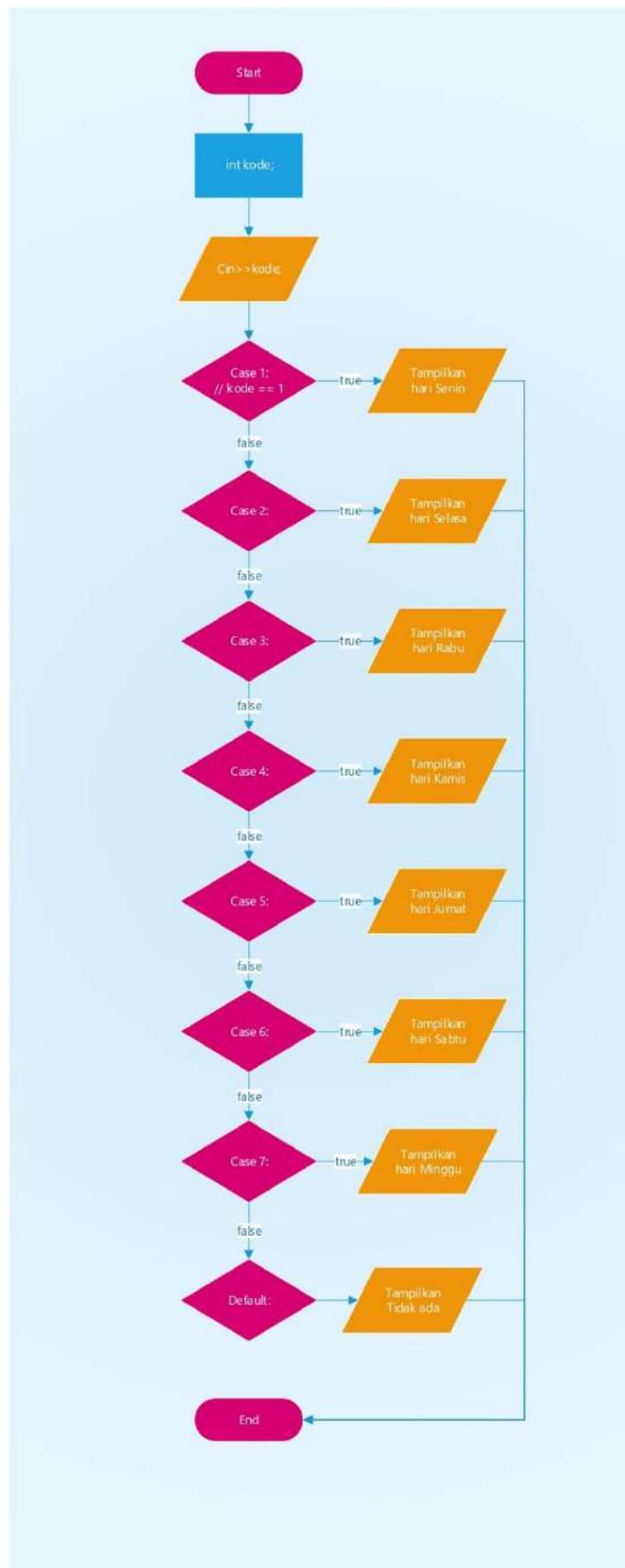
1. Jelaskan kelebihan Perintah Switch dibandingkan Perintah IF!
Switch digunakan untuk persamaan bukan perbandingan.
2. Mengapa anda sebagai programmer harus menggunakan perintah Switch dalam program!
Karena penggunaan switch sangat tepat untuk menangani pengambilan keputusan. Contoh: switch lebih baik digunakan dari pada if bertingkat.
3. Sebutkan dan jelaskan perintah - Perintah penjukung switch!
case berfungsi sebagai Pemilihannya.
break berfungsi untuk mengakhiri eksekusi switch
default berfungsi jika tidak ada case yg cocok.
4. Buatlah contoh algoritma dan program sederhana menggunakan perintah switch dengan memakai flowchart!

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 screenshot shows a code editor window titled "pertemuan 3 laporan akhir tugas akhir.cpp" containing C++ code. The code defines a function main() that prompts the user for a day code (1-7) and prints the corresponding day name. A terminal window to the right shows the execution of the program, where the user inputs '1' and the output is 'Senin'. The code uses a switch statement with break statements after each case.

```
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 }
```

D:\Documents\File Dev C++\pertemuan 3 laporan akhir tug... — □ X
Masukkan kode: 1
Senin

Process exited after 1.13 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

The image shows a C++ development environment with a code editor on the left and five terminal windows on the right, each showing the execution of a program named 'pertemuan 4 akhir.cpp'.

Code Editor Content:

```
1 #include<iostream>
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 (100000 <= harga_beli && harga_beli < 200000)
11    {
12        cout<<"Discount 5%";
13    }
14    else if (200000 <= harga_beli && harga_beli < 300000)
15    {
16        cout<<"Tiket ke Yogyakarta";
17    }
18    else if (300000 <= harga_beli && harga_beli < 400000)
19    {
20        cout<<"Tiket ke Bali";
21    }
22    else if (400000 <= harga_beli && harga_beli < 500000)
23    {
24        cout<<"Jam Tangan Rolex";
25    }
26    else if (harga_beli > 500000)
27    {
28        cout<<"Tiket ke Swiss";
29    }
30    else
31    {
32        cout<<"Coba Lagi"<<endl;
33    }
34
35    return 0;
36 }
```

Terminal Outputs:

- Masukkan Harga Pembelian: 150000
Discount 5%
- Masukkan Harga Pembelian: 225000
Tiket ke Yogyakarta
- Masukkan Harga Pembelian: 355000
Tiket ke Bali
- Masukkan Harga Pembelian: 465000
Jam Tangan Rolex
- Masukkan Harga Pembelian: 600000
Tiket ke Swiss

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

Praktium Algoritma
Laporan Awal - Pertemuan 5

TEORI DASAR

a) Pendahuluan

Perintah pengulangan (looping) digunakan untuk melaksanakan proses berulang-ulang. Jika suatu kondisi sudah terpenuhi maka perulangan diberhentikan.

Proses perulangan biasanya digunakan untuk proses pemrosesan data, mengulang proses perhitungan dan mengulang proses penampilan hasil pengolahan data.

b) FOR

Proses looping yg menggunakan perhitungan (counter) saat dibuat dg pernyataan for. Pernyataan ini digunakan bila anda sudah tahu berapa kali anda akan mengulang satu atau beberapa pernyataan.

Bentuk umum for:

for (unguapan1 i = kendala; unguapan2)

{
pernyataan;
}

1. Unguapan 1 adalah inisialisasi terhadap variable pengontrol looping

2. Keduaan adalah kondisi untuk keluar looping

3. Unguapan2 adalah Pengaturan kenaikan atau penurunan nilai variable pengontrol looping.

Tugas Pendahuluan

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 dalam beberapa hal, misalkan mengulang proses memasukkan data, mengulang proses perhitungan, dan mengulang proses penampiran hasil pengolah 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.

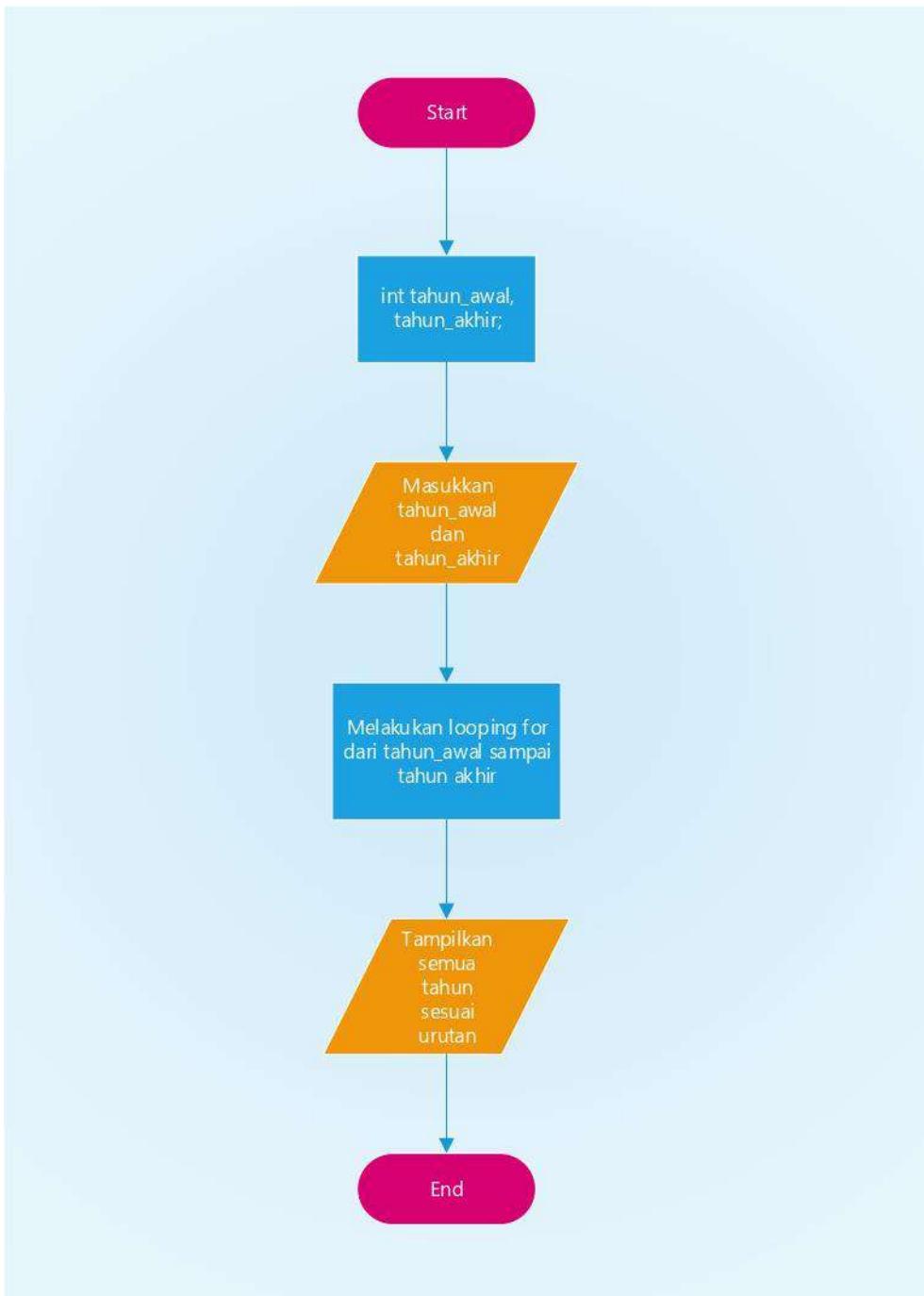
4. Buatlah contoh algoritma dan program sederhana menggunakan perintah for > dg memakai flowchart!

Jawaban No. 4

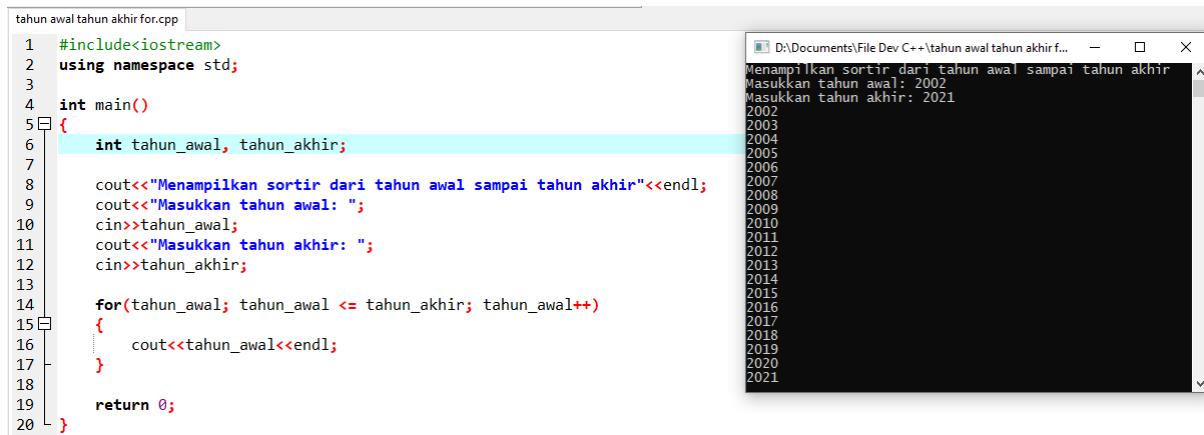
Algoritma

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

Flowchart



Contoh Program:



The screenshot shows a C++ development environment. On the left, the code editor displays a file named 'tahun_awal_tahun_akhir_for.cpp' containing a program to print years from a start to an end year. On the right, a terminal window titled 'D:\Documents\File Dev C++\tahun_awal_tahun_akhir f...' shows the output of the program, which asks for input and then prints years from 2002 to 2021.

```
tahun_awal_tahun_akhir_for.cpp
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 }
```

```
D:\Documents\File Dev C++\tahun_awal_tahun_akhir_for.cpp
Menampilkan sortir dari tahun awal sampai tahun akhir
Masukkan tahun awal: 2002
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
```

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 screenshot shows a code editor with two tabs: "tugas akhir - pertemuan 6.cpp" and "tugas akhir - pertemuan 5.cpp". The "tugas akhir - pertemuan 5.cpp" tab contains the following C++ code:

```
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 }
```

To the right of the code editor is a terminal window titled "D:\Documents\File Dev C++\tugas akhir - pertemuan 5.exe". It displays the output of the program:

```
Masukkan digit maksimal fibonacci: 20
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181
Process exited after 1.638 seconds with return value 0
Press any key to continue . . .
```

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 pengkodisan loop tetap dilakukan atau berhenti

pernyataan 3 = untuk menentukkan 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

Laporan Awal-Pertemuan 6

Teori Dasar

a) Do... While

Perintah ini menyatakan Pengulangan Proses selama kondisi tertentu. Mengerjakan do terlebih dahulu, kemudian melakukan Pengujian pada while.

Bentuk umum do while:

do

 Pernyataan;

 while (kondisi);

 atau

 do

 {

 Pernyataan;

 Pernyataan;

 }

 while (kondisi);

b) While

Perintah while ini prinsipnya sama dg perintah do...while hanya pengujian kondisinya terletak pada loop. Langsung melakukan pengujian.

Bentuk umum while:

while (kondisi)

 Pernyataan;

 atau

 while (kondisi)

 {

 Pernyataan;

 Pernyataan;

 }

Pernyataan dapat berupa pernyataan tunggal atau beberapa pernyataan yg dibatasi dg tanda { }. Pernyataan ini dijalankan bila kondisinya benar.

TUGAS PENGETAHUAN

1. Mengapa anda sebagai 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: dapat menjalankan kode program tanpa harus mengetik ulang.
Kekurangan: tidak bisa melakukna kode program yg berbeda
3. Buatlah Contoh algoritma dan program sederhana menggunakan perintah do...while dgn memauai flowchart!
4. Buatlah Contoh algoritma dan program sederhana menggunakan perintah while dgn memauai flowchart!

Jawaban No.3 dan 4

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

```
tugas awal pertemuan 6.cpp
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 }
```

D:\Documents\File Dev C++\tugas awal pertemuan 6.... — X

```
Masukkan nama: Andri
Halo, Andri
Lagi? Y/n
Jawab: y
Masukkan nama: Firman
Halo, Firman
Lagi? Y/n
Jawab: y
Masukkan nama: Saputra
Halo, Saputra
Lagi? Y/n
Jawab: n
```

Process exited after 8.256 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;
```

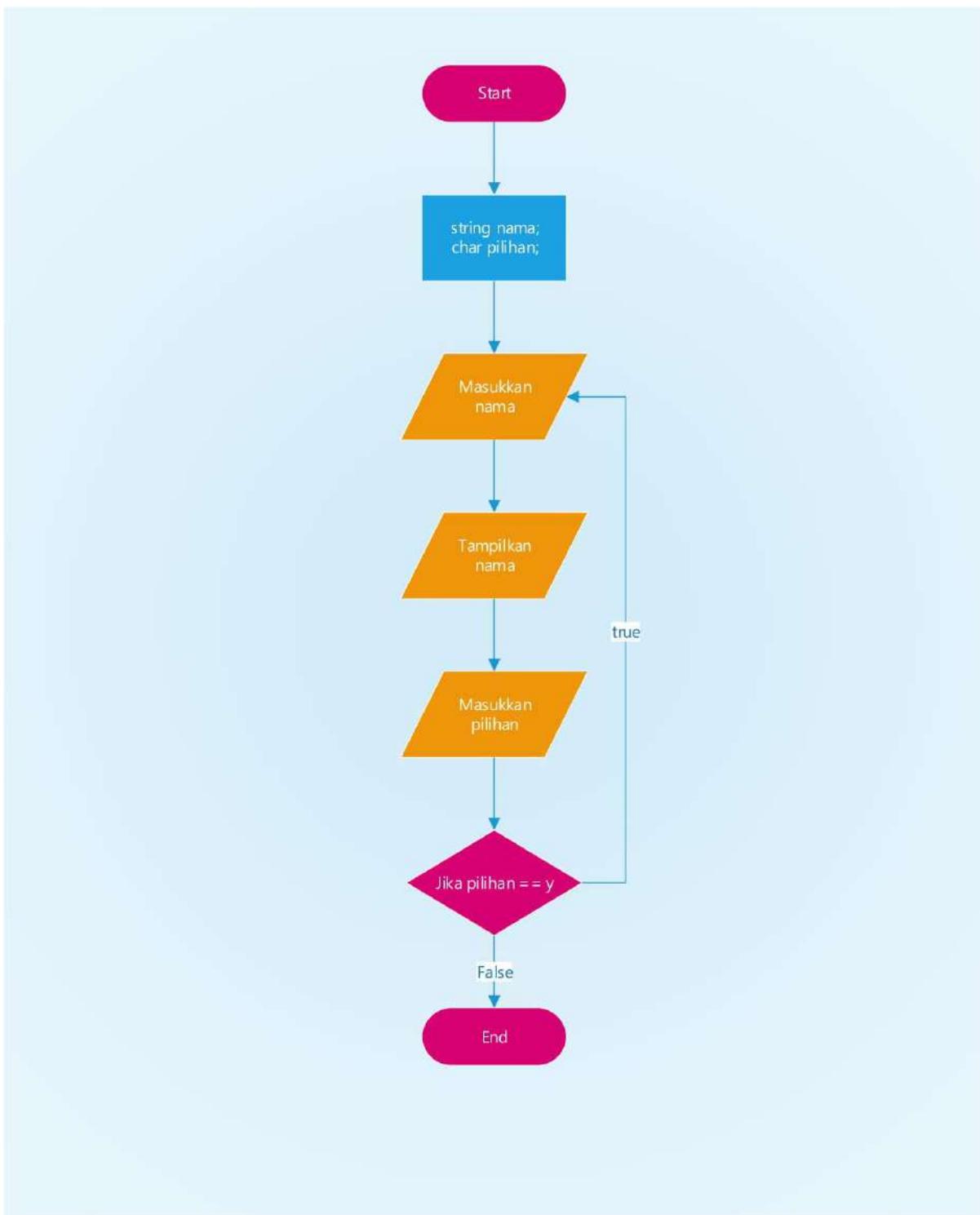
```
do
{
    cout<<"Masukkan nama: ";
    cin>>nama;
    cout<<"Halo, "<<nama<<endl;

    cout<<"Lagi? Y/n"<<endl;
    cout<<"Jawab: ";
    cin>>pilihan;

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

return 0;
}
```

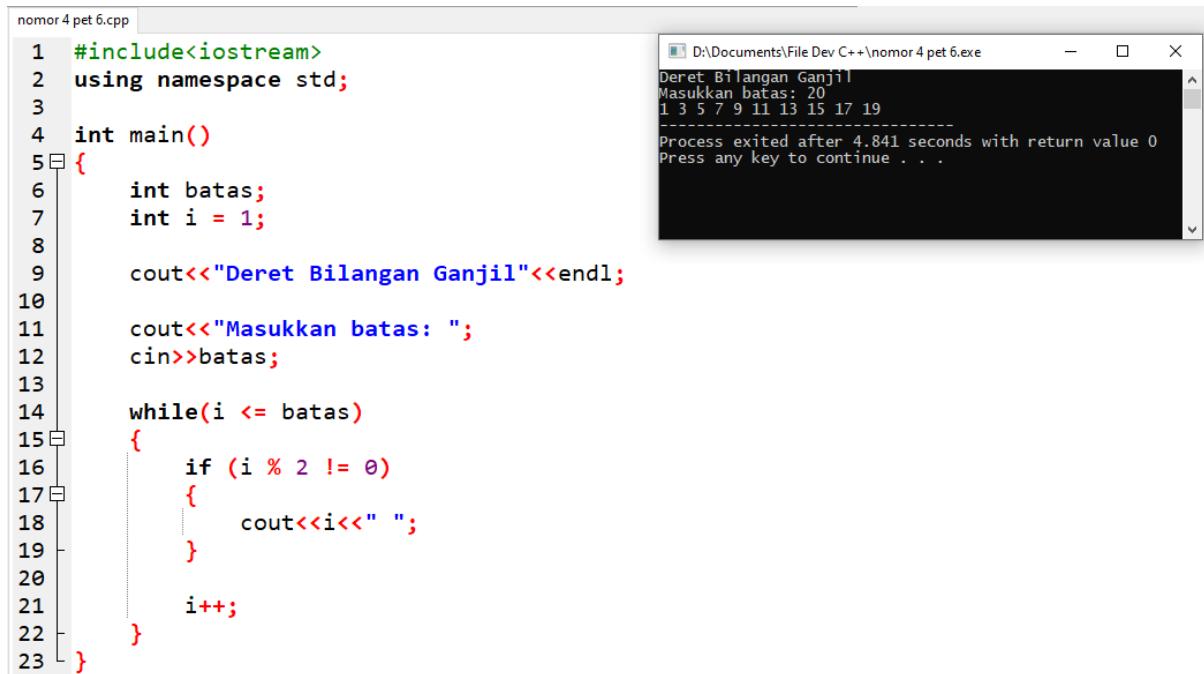
Flowchart



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 screenshot shows a C++ development environment. On the left, the code editor displays a file named "nomor 4 pet 6.cpp" containing the following code:

```
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 }
```

On the right, the terminal window shows the program's output:

```
D:\Documents\File Dev C++\nomor 4 pet 6.exe
Deret Bilangan Ganjil
Masukkan batas: 20
1 3 5 7 9 11 13 15 17 19
Process exited after 4.841 seconds with return value 0
Press any key to continue . . .
```

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<<" ";

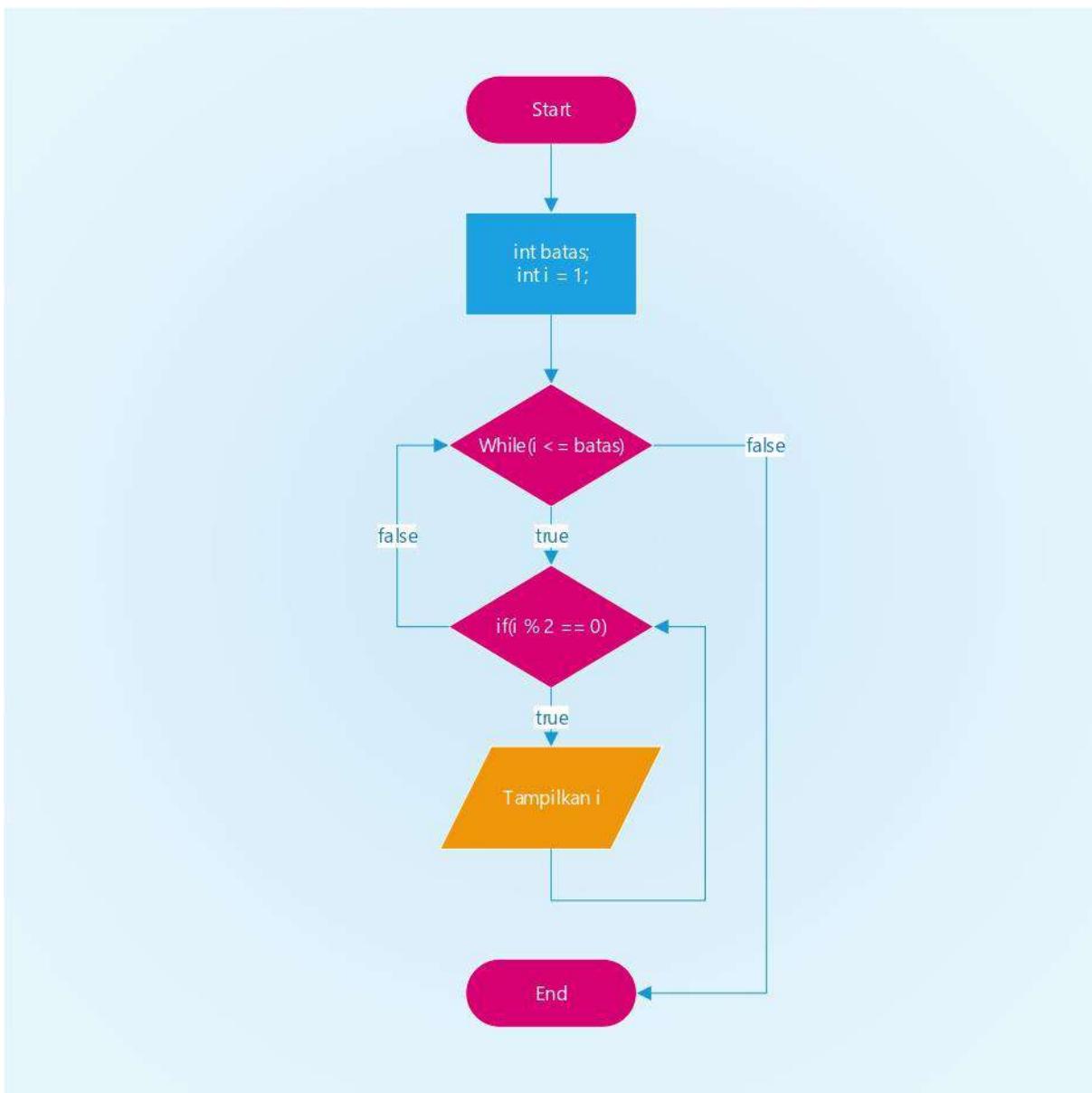
    }

    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++ development environment with two windows. The left window is a code editor titled "tugas akhir - pertemuan 6.cpp" containing the following C++ 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<<a;
18        c = a + b;
19        a = b;
20        b = c;
21        i++;
22    }
23    return 0;
24 }
```

The right window is a terminal or command prompt showing the output of the program. It displays the input "Masukkan digit maksimal fibonacci: 15" followed by the output "0112358132134558914423377". Below the output, it says "Process exited after 1.092 seconds with return value 0" and "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 Saputra

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 balik)

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 :

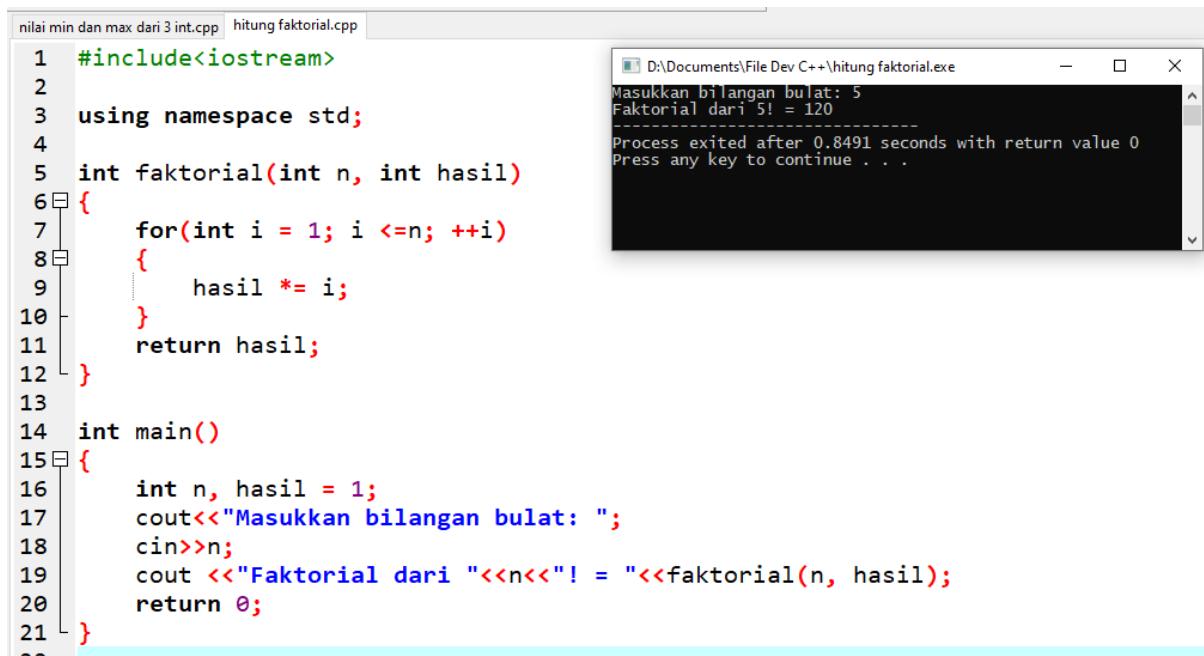
nama_fungsi (parameter jika ada);

Tugas Pendahuluan

1. Jelaskan yg dimaksud dg 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 penjukung fungsi!
 - tipe_keluaran = jenis tipe data yg dikeluarkan / diembalikan . 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 memauai flowchart!

Jawaban No. 4

Contoh Program



The screenshot shows a code editor with two tabs: "nilai min dan max dari 3 int.cpp" and "hitung faktorial.cpp". The "hitung faktorial.cpp" tab is active, displaying the following C++ code:

```
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 }
```

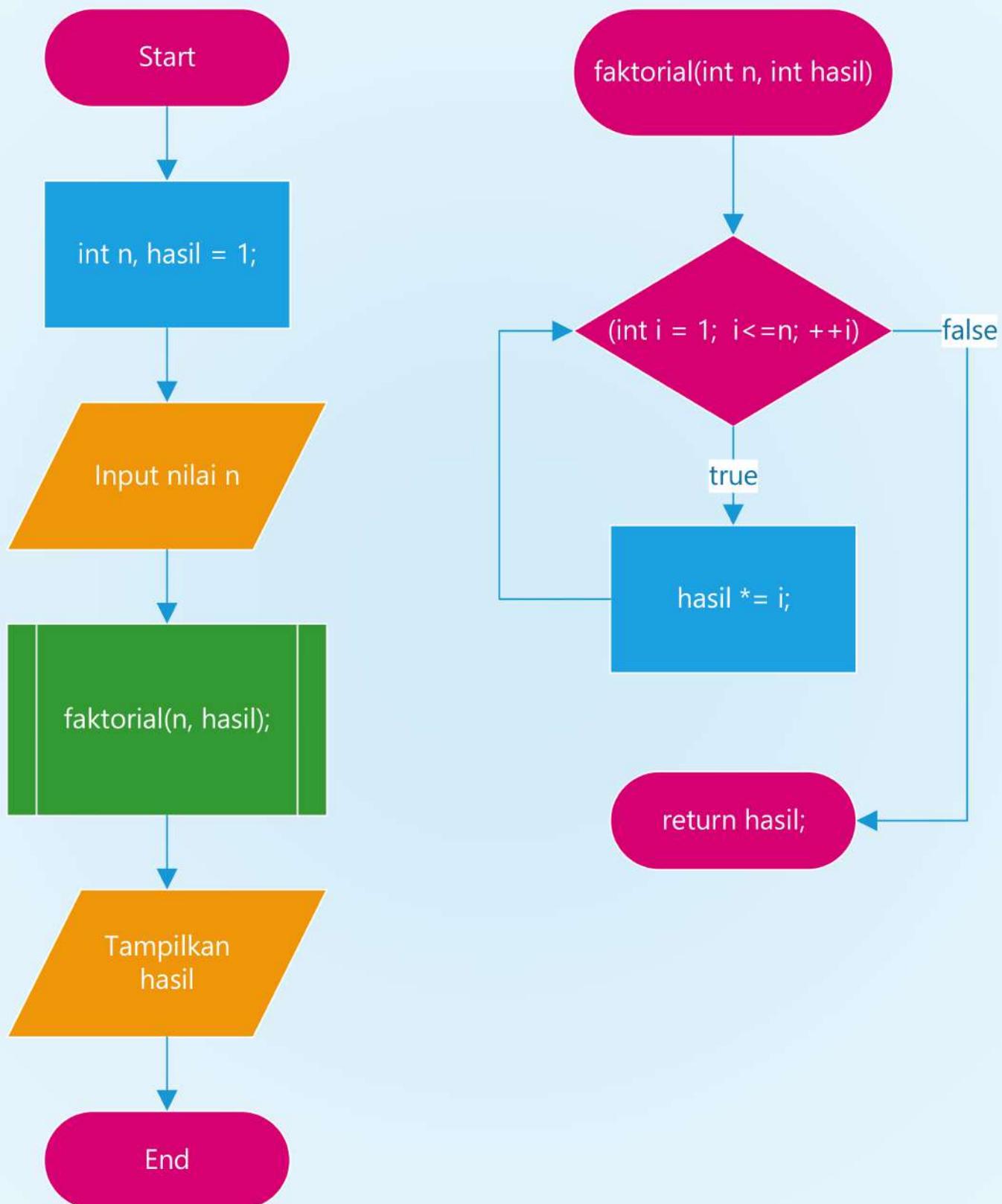
To the right of the code editor is a terminal window titled "D:\Documents\File Dev C++\hitung faktorial.exe". It displays the following output:

```
Masukkan bilangan bulat: 5
Faktorial dari 5! = 120
Process exited after 0.8491 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

The screenshot shows a C++ development environment with two panes. The left pane displays the source code file 'genap_ganjil_function.cpp' containing C++ code. The right pane shows the terminal window where the program is run.

```
genap_ganjil_function.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_function.exe

```
Masukkan angka: 15
Ganjil
-----
Process exited 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: Andri Firman Saputra

Praktikum Algoritma

NIM : 201011402125

Laporan Awal - Pertemuan 2

Teori Dasar

a) Pendektauan

Array merupakan kolesi data dimana setiap elemen mempunyai nama dan tipe yg sama serta setiap elemen diaulus dgn membedakan indeks array-nya.

Pendektaasian:

```
int C [7];
```

inisialisasi nilai pada saat pendektaasian:

```
int C [7] = {-45, 0, 6, 72, 1543, 43, 47};
```

b) Array dimensi satu

Bentuknya:

```
Tipe_Data nama_var [Ukuran];
```

c) Array dimensi dua

Bentuknya:

```
Tipe_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:

```
Tipe_Data nama_var [Ukuran][Ukuran][Ukuran], dst;
```

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

Contoh:

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

TUGAS Pendahuluan

1. Jelaskan yg dimaksud dgn ARRAY!

ARRAY merupakan kolesi data dimana setiap elemen mempunyai 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 utp[2][5][7];

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

- Mudah digunakan Untuk menampung data dgn tipe data dan nama Variabel yg sama. Contoh: String nama_mhs[100];
- Mudah Untuk Menampilkan jumlah data yg banyak:

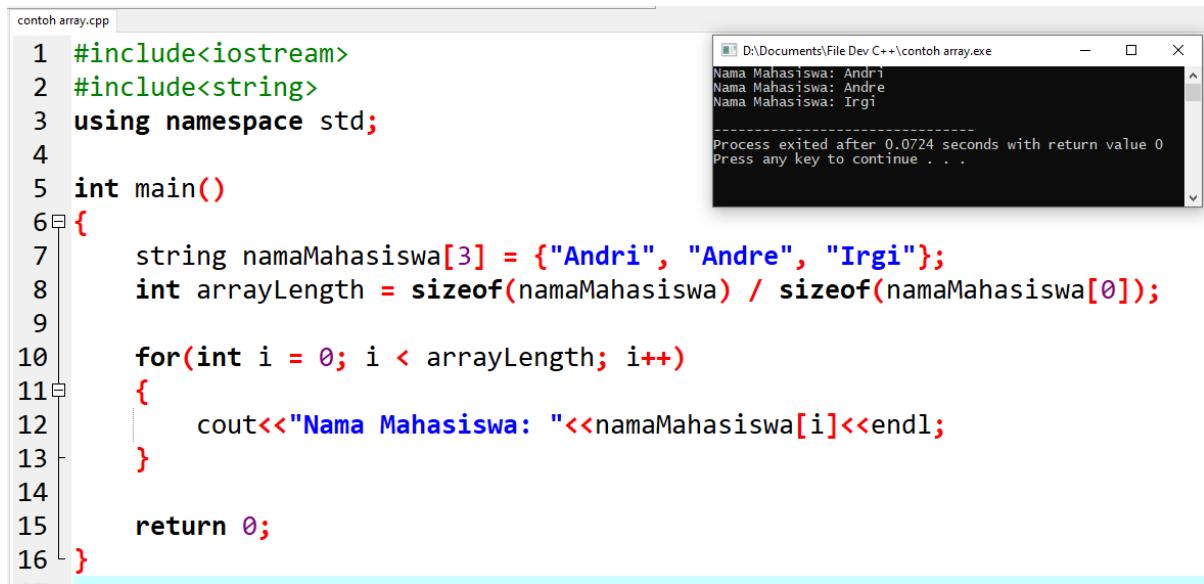
```
for(i=0; i<100; i++)  
{  
    cout << nama_mhs[i];  
    cout << endl;  
}
```

- Efisiensi yg tinggi
- tidak berantauan

4. Buatlah Contoh algoritma dan program Sederhana Menggunakan array dgn Memakai flowchart!

Jawaban No. 4

Contoh Program:



The screenshot shows a code editor and a terminal window. The code editor contains the following C++ code:

```
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 }
```

The terminal window shows the program's output:

```
D:\Documents\File Dev C++\contoh array.exe
Nama Mahasiswa: Andri
Nama Mahasiswa: Andre
Nama Mahasiswa: Irgi
Process exited after 0.0724 seconds with return value 0
Press any key to continue . . .
```

Source Code:

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

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

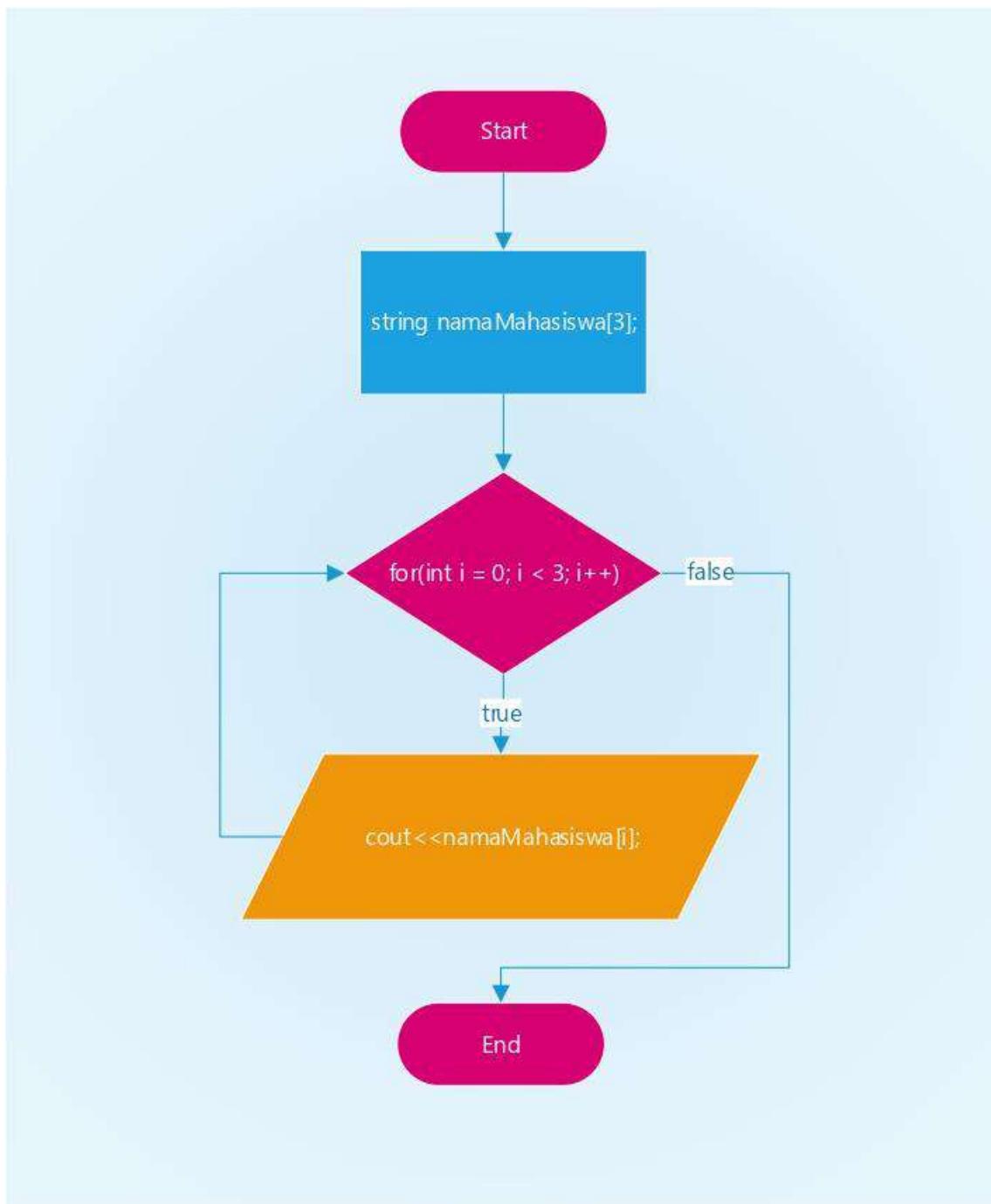
    for(int i = 0; i < arrayLength; i++)
    {
        cout<<"Nama Mahasiswa: "<<namaMahasiswa[i]<<endl;
    }

    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++ development environment with two windows. The left window is titled "matrิกs 3x3.cpp" and contains the following code:

```
matrิกs 3x3.cpp
1 #include<iostream>
2 #include<iomanip>
3 using namespace std;
4
5 int main()
6 {
7     int a[3][3], b[3][3], c[3][3], i, j, k;
8
9     cout<<"Matriks 3 x 3 : "<<endl;
10
11    for(i=0;i<3;i++)
12    {
13        for(j=0;j<3;j++)
14        {
15            cout<<"Input Baris "<<(i+1)<<, Kolom "<<(j+1)<< "=";
16            cin>>a[i][j];
17        }
18        cout<<endl;
19    }
20
21    cout<<"Matriks 3 x 3 = "<<endl<<endl;
22
23    for(i=0;i<3;i++)
24    {
25        for(j=0;j<3;j++)
26        {
27            cout<<setw(4)<<a[i][j];
28        }
29        cout<<endl<<endl;
30    }
31 }
```

The right window is titled "D:\Documents\File Dev C++\matriks 3x3.exe" and shows the output of the program. It prompts for a 3x3 matrix and displays the input and resulting matrix:

```
D:\Documents\File Dev C++\matriks 3x3.exe
Matriks 3 x 3 :
Input Baris 1 , Kolom 1 = 5
Input Baris 1 , Kolom 2 = 3
Input Baris 1 , Kolom 3 = 9
Input Baris 2 , Kolom 1 = 2
Input Baris 2 , Kolom 2 = 1
Input Baris 2 , Kolom 3 = 4
Input Baris 3 , Kolom 1 = 7
Input Baris 3 , Kolom 2 = 9
Input Baris 3 , Kolom 3 = 8
Matriks 3 x 3 =
5   3   9
2   1   4
7   9   8
-----
Process exited after 18.83 seconds with return value 0
Press any key to continue . . .
```

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 biasa dipakai dalam Pemrograman guna menampung dan memanipulasi data teks.

Di dalam turbo C++, string bukan merupakan tipe data tersendiri tetapi hanya merupakan jenis khusus dari array.

String juga bisa dipergunakan dalam konstanta dan variabel.

b) Fungsi pada String

Pada Turbo C++ disediakan sejumlah fungsi Mauro yg didefinisikan di preprosesor #define Mauro - Mauro yg didefinisikan di type.h adalah:

1) isalnum()

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

Bentuk umum :

```
tipe-data isalnum( tipe argument );
```

2) isalpha()

Fungsi Mauro ini akan menghasilkan boolean true, kalau argumennya adalah huruf bukan angka.

Bentuk umum :

```
tipe-data isalpha( tipe argument );
```

3) isdigit()

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

Bentuk umum :

```
tipe-data isdigit( tipe argument );
```

4) islower()

Fungsi Mauro ini akan menghasilkan boolean true, kalau argumennya adalah huruf kecil.

Bentuk umum :

```
tipe-data islower( tipe argument );
```



5) isupper()

Fungsi Mauro ini akan menghasilkan boolean true, kala argumennya adalah huruf besar.

Bentuk umum:

```
tipe_data isupper (tipe argument);
```

6) tolower()

Fungsi Mauro ini akan menghasilkan huruf kecil jika diisi huruf besar.

Bentuk umum:

```
tipe_data tolower (tipe argument);
```

7) toupper()

Fungsi Mauro ini kebalikan dari tolower, menghasilkan huruf Besar.

Bentuk umum:

```
tipe_data toupper (tipe argument);
```

8) strcpy()

Fungsi ini dipakai untuk menyalin string asal ke variabel string tujuan harus memiliki ukuran yg dapat menampung 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 bagian akhir string tujuan.

Bentuk umum:

```
strcat( tujuan, asal);
```



11) strcmp()

Fungsi ini dipakai untuk membandingkan str1 dengan str2.

Nilai hasil fungsi ini boolean true.

Jadi huruf 'A' lebih kecil dari 'B' dst.

12.) strlwr()

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

Contoh:

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

```
strlwr(nama);
```

```
hasilnya: andri
```

13) strupr()

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

Contoh:

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

```
strupr(nama);
```

```
hasil: ANDRI
```

TUGAS Pendahuluan

1. Jelaskan jenis-jenis fungsi string pada bahasa Pemrograman C/C++!
isalnum(), isalpha(), isdigit(), islower(), isupper(), tolower(), toupper(), strcpy(), strlen(), strcat(), strcmp(), strlwr(), strupr()
2. Sebagai programmer, Mengapa anda menggunakan fungsi string?
Karena fungsi string memiliki banyak kelebihan, seperti
untuk melakukan Perbandingan, Merapikan string, memindahkan
string, menggabungkan string.
3. Buatlah Contoh Program sederhana menscopy beberapa karakter
menggunakan array!
4. Buatlah Contoh algoritma dan program sederhana menggunakan
Salah satu fungsi String untuk menghitung jumlah karakter
dgn memakai flowchart!



3.

The screenshot shows a code editor window titled "strcpy.cpp" containing C++ code. The code includes #include directives for iostream and string.h, uses namespace std, defines a main function, initializes a character array "nama" with the value "Andri Firman Saputra", creates another character array "destinasi", copies the contents of "nama" to "destinasi" using strcpy, and then prints both arrays to the console using cout. The output window shows the program's execution path, the copied values, and a message indicating the process exited successfully.

```
1 #include<iostream>
2 #include<string.h>
3 using namespace std;
4
5 int main()
6 {
7     char nama[25] = {"Andri Firman Saputra"};
8     char destinasi[25];
9
10    strcpy(destinasi, nama);
11    cout<<"Nama\t\t: "<<nama<<endl;
12    cout<<"Destinasi\t: "<<destinasi;
13
14    return 0;
15 }
```

D:\Documents\File Dev C++\strcpy.exe

Nama : Andri Firman Saputra
Destinasi : Andri Firman Saputra

Process exited after 0.0596 seconds with return value 0

Press any key to continue . . .

Source Code:

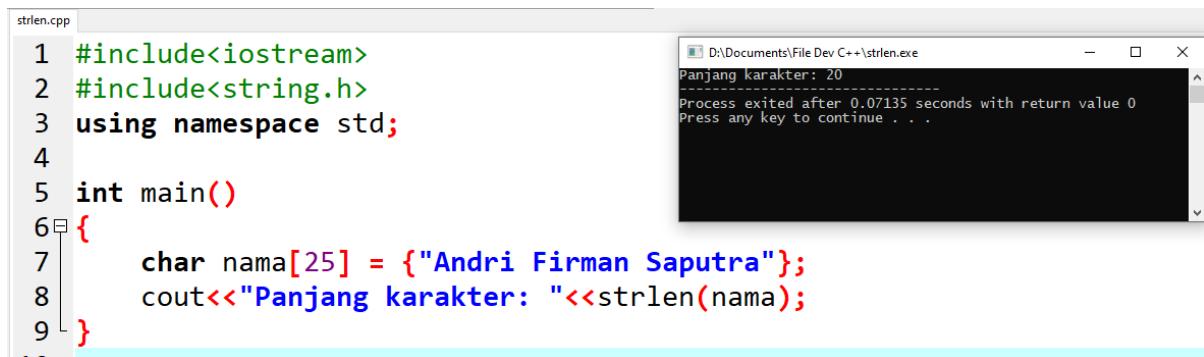
```
#include<iostream>
#include<string.h>
using namespace std;

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

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

    return 0;
}
```

4.



```
strlen.cpp
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 }
```

D:\Documents\File Dev C++\strlen.exe
Panjang karakter: 20
Process exited after 0.07135 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 screenshot shows a code editor window titled 'akhir pertemuan 9.cpp' containing C++ code. The code prompts the user for their name and hobby, then prints them back along with a computer answer. To the right is a terminal window titled 'D:\Documents\File Dev C++\akhir pertemuan 9.exe' showing the program's output after it has been run.

```
akhir pertemuan 9.cpp
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 }
```

```
D:\Documents\File Dev C++\akhir pertemuan 9.exe
Masukkan Nama: Andri Firman Saputra
Masukkan Hobi: Mengoding
Jawaban Komputer
Nama: Andri Firman Saputra
Hobi: Mengoding
-----
Process exited after 7.828 seconds with return value 0
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

Praaktikum Algoritma
Laporan Awal - Pertemuan 1a

Teori Dasar

a) Pendahuluan

Pointer adalah variabel yg berisi alamat memori sebagai penilaianya 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++:

tipe-data *variabel name;

* adalah operator memori yg fungsinya untuk mengembalikan 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 jari sebuah memori berdasarkan alamat memori.

Tugas Pendahuluan

1. Jelaskan yg dimaksud dgn pointer!

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

2. Sebutkan dan jelaskan jenis-jenis pointer!

- Pointer aritmetik

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

- pointer array

pointer dapat di-array seperti tipe data lainnya dalam c++.

Contoh:

int *pi[10];

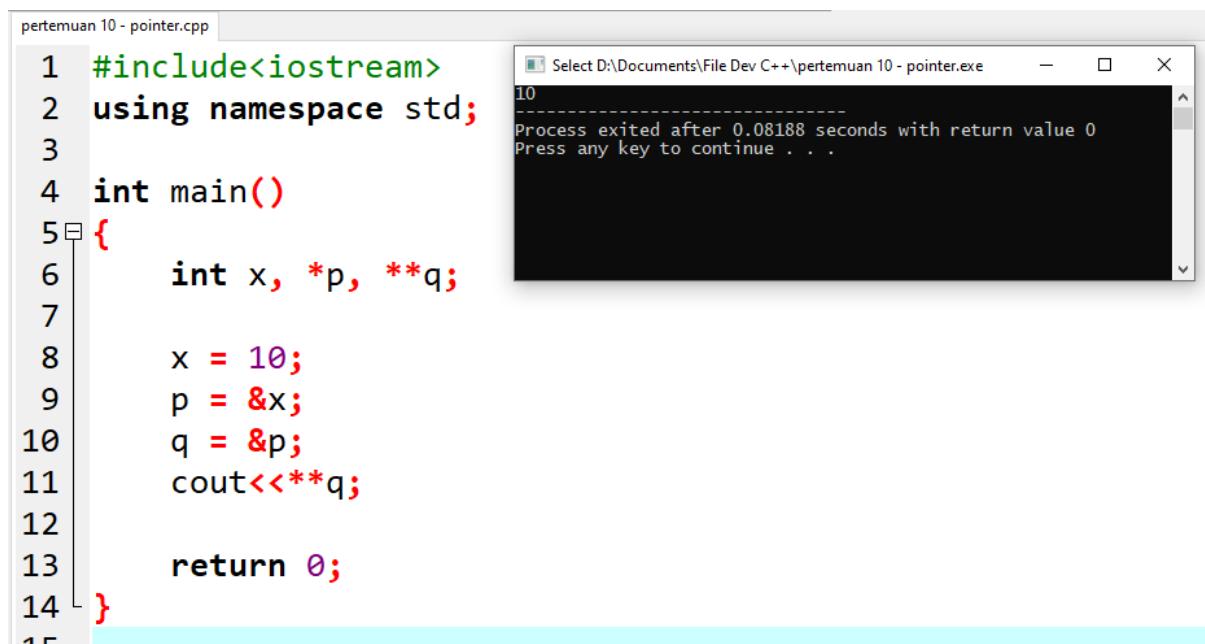


3. Sebagai programmer, kapan anda menggunakan pointer)
saat memindahkan data dari satu fungsi ke fungsi lainnya
dgn mengirim addressnya saja untuk mendefinisi nilai.

4. Buatlah contoh algoritma dan program sederhana menggunakan
pointer dgn memakai flowchart!



Jawaban No. 4



The screenshot shows a C++ development environment. On the left, a code editor window titled "pertemuan 10 - pointer.cpp" displays the following 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 }
```

On the right, a terminal window titled "Select D:\Documents\File Dev C++\pertemuan 10 - pointer.exe" shows the output of the program:

```
10
Process exited after 0.08188 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



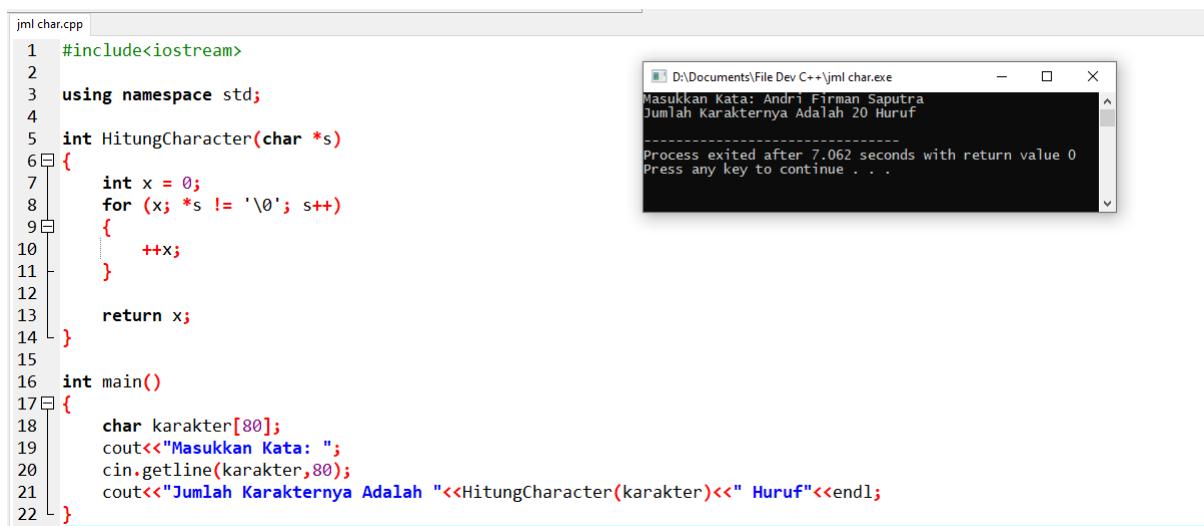
Disusun Oleh:

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



The screenshot shows a code editor with a file named "jml char.cpp". The code contains a function to count characters in a string and a main function that prompts the user for input and prints the character count. To the right, a terminal window displays the output of the program when run.

```
jml char.cpp
1 #include<iostream>
2
3 using namespace std;
4
5 int HitungCharacter(char *s)
6 {
7     int x = 0;
8     for (x; *s != '\0'; s++)
9     {
10         ++x;
11     }
12
13     return x;
14 }
15
16 int main()
17 {
18     char karakter[80];
19     cout<<"Masukkan Kata: ";
20     cin.getline(karakter,80);
21     cout<<"Jumlah Karakternya Adalah "<<HitungCharacter(karakter)<<" Huruf"<<endl;
22 }
```

D:\Documents\File Dev C++\jml char.exe
Masukkan Kata: Andri Firman Saputra
Jumlah Karakternya Adalah 20 Huruf
Process exited after 7.062 seconds with return value 0
Press any key to continue . . .

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 penilaianya 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 di kelompokan kedalam satu nama. (menurut Pascal struktur juga dikenal sebagai record). Struktur membantu data-data yg rumit, khususnya dalam program yg besar, karen 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 dibangkitkan (derived data type), yg disusun menggunakan objek dg 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 di terapkan pada struktur adalah proses menyalin atau menunjukkan 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 value 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 tipe. Untuk menghubungkan 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 menunjukkan kepada sebuah data yg mempunyai struktur PELAJAR seperti berikut:

Struct PELAJAR *ptPelajar;

seperti pada pointer yg lain, dekarasi di atas 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 dg menggunakan symbol \rightarrow seperti berikut ini:

PtPelajar \rightarrow nama = Khairul Ariffin;

PtPelajar \rightarrow kelas = 12345;

PtPelajar \rightarrow faultas = Ekonomi;

SAMA dengan :

* PtPelajar.nama = Khairudin;

* PtPelajar.kelas = 12345;

* PtPelajar.faultas = Ekonomi;

Tugas Pendahuluan

1. Jelaskan yg dimaksud dg 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 lanjut 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 sejernih menggunakan struktur dg memanusi flowchart!

Jawaban No. 4

The screenshot shows a code editor window titled "pertemuan 11.cpp" and a terminal window titled "D:\Documents\File Dev C++\pertemuan 11.exe". The code in the editor is a C++ program that defines a struct for movie information, reads user input for two movies, and prints them out. The terminal window shows the program's execution, including prompts for movie titles and years, and the resulting output.

```
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 } mine, yours;
12
13
14 void printfmovie(movies_t movie);
15
16 int main()
17 {
18     char buffer[100];
19     strcpy(mine.title, "Borat: Cultural Learnings of America for Make Benefit Glorious Nation of Kazakhstan");
20     mine.year = 2006;
21     cout<<"Masukkan judul favorit: ";
22     cin>>yours.title;
23     cout<<"Masukkan tahun: ";
24     cin>>buffer;
25     yours.year = atoi(buffer);
26     cout<<"Judul film favorit yang ada:\n";
27     printfmovie(mine);
28     cout<<"Judul film favorit kamu adalah:\n";
29     printfmovie(yours);
30
31     return 0;
32 }
33
34 void printfmovie(movies_t movie)
35 {
36     cout<<movie.title;
37     cout<<"("<<movie.year<<")\n";
38 }
```

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

The screenshot shows a code editor on the left with the file 'data_karyawan.cpp' containing C++ code, and a terminal window on the right displaying the program's output.

Source Code:

```
1 #include<iostream>
2 #include<string>
3 using namespace std;
4
5 int main()
6 {
7     int jml_karyawan, gaji_per_bulan, jml_anak;
8     char status_menikah;
9     string nama;
10    cout<<"\t Data Karyawan" << endl;
11    cout<<"\t -----" << endl;
12    cout<<"\t Masukkan jumlah karyawan \t: ";
13    cin >> jml_karyawan;
14    cout<<"\t -----" << endl;
15
16    for(int i = 0; i < jml_karyawan; i++)
17    {
18        cout << "\t Karyawan ke-" << i+1 << endl;
19        cout << "\t Masukkan nama \t\t\t: ";
20        // ws = whitespace
21        getline(cin >> ws, nama);
22        cout << "\t Masukkan gaji per bulan \t: ";
23        cin >> gaji_per_bulan;
24        cout << "\t Masukkan status menikah (s/b) \t: ";
25        cin >> status_menikah;
26        if(status_menikah == 's')
27        {
28            cout << "\t Masukkan jumlah anak \t\t: ";
29            cin >> jml_anak;
30        }
31        else
32        {
33            jml_anak = 0;
34        }
35    }
36 }
```

Output:

```
D:\Documents\File Dev C++\data_karyawan.exe
Data Karyawan
Masukkan jumlah karyawan : 3
Karyawan ke-1
Masukkan nama : Andre Firman Saputra
Masukkan gaji per bulan : 50000000
Masukkan status menikah (s/b) : s
Masukkan jumlah anak : 10
Karyawan ke-2
Masukkan nama : Andre Farhan Saputra
Masukkan gaji per bulan : 4500000
Masukkan status menikah (s/b) : b
Masukkan nama : Muhammad Irgi Al-ghitraf
Masukkan gaji per bulan : 3500000
Masukkan status menikah (s/b) : s
Masukkan jumlah anak : 2

Process exited after 41.24 seconds with return value 0
Press any key to continue . . .
```

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 kelompokan 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 dibangkitkan (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

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 sebarang dgn elemen berikutnya. Pembandingan aramatanya dapat dimulai dari data yg paling awal atau yg paling akhir.

Jika elemen yg sebarang (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 sebarang dgn elemen berikutnya sampai elemen terakhir.

Jika ditemukan elemen lain yg lebih kecil dari elemen sebarang 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 dg n elemen yg lain, apabila nilai nya lebih kecil maka datanya ditampung di elemen yg lain lagi.

Tugas Pendahuluan

1. Apa yg dl maksud dg sorting!

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

2. Sebutan 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 berantauan dan sulit dibaca, sebaliknya jika kita menggunakan sorting data lebih mudah dibaca dan diolah.

4. Buatlah contoh algoritma dan program sederhana menggunakan sorting dg memakai flowchart!

Contoh Program Sorting:

The screenshot shows a C++ development environment with two windows. On the left is the code editor window titled "order asc.cpp" containing the following C++ code:

```
order asc.cpp
1 #include<iostream>
2 using namespace std;
3
4 void BubbleSort(int data[], int n);
5
6 int main()
7 {
8     int i, n = 9, data[] = {10, 40, 20, 60, 15, 4, 8, 100, 200, 25};
9
10    cout<<"Pengurutan Menaik"<<endl;
11    cout<<"-----"<<endl;
12    cout<<"Sebelum diurutkan: "<<endl;
13
14    for(i = 0; i <= n; i++)
15    {
16        cout<<data[i]<<" ";
17    }
18
19    cout<<endl;
20
21    cout<<"-----"<<endl;
22
23    BubbleSort(data, n);
24
25    cout<<"Setelah diurutkan: "<<endl;
26
27    for(i = 0; i <= n; i++)
28    {
29        cout<<data[i]<<" ";
30    }
31
32    cout<<endl;
33
34    return 0;
35 }
36
37 void BubbleSort(int array1[], int n)
38 {
39     int i, j, tmp;
40
41     for(i = 1; i <= n; i++)
42     {
43         for(j = n; j >= i; j--)
44         {
45             // controller asc and desc
46             if(array1[j] < array1[j-1])
47             {
48                 tmp = array1[j];
49
50                 array1[j] = array1[j-1];
51
52                 array1[j-1] = tmp;
53             }
54         }
55     }
56 }
```

The right window shows the terminal output of the program. It starts with "Pengurutan Menaik", followed by a separator line, and then "Sebelum diurutkan:" followed by the initial array elements: 10 40 20 60 15 4 8 100 200 25. Another separator line follows, then "Setelah diurutkan:" followed by the sorted array elements: 4 8 10 15 20 25 40 60 100 200. Below the output, it says "Process exited after 0.05929 seconds with return value 0" and "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

The screenshot shows a development environment with two windows. On the left is a code editor titled "pertemuan 12 - akhir.cpp" containing C++ code for bubble sort. On the right is a terminal window titled "D:\Documents\File Dev C++\pertemuan 12 - akhir.exe" showing the execution of the program.

```
1 #include<iostream>
2 using namespace std;
3
4 void BubbleSort(int data[], int n);
5
6 int main()
7 {
8     int i,
9     n = 9,
10    data[] = {4, 8, 5, 9, 6, 2, 7, 5, 9, 5};
11
12    cout<<"Pengurutan Menurun"=<endl;
13    cout<<"-----"=<endl;
14    cout<<"Sebelum diurutkan: "<<endl;
15
16    for(i = 0; i <= n; i++)
17    {
18        cout<<data[i]<<" ";
19    }
20
21    cout<<endl;
22
23    cout<<"-----"=<endl;
24
25    BubbleSort(data, n);
26
27    cout<<"Setelah diurutkan: "<<endl;
28
29    for(i = 0; i <= n; i++)
30    {
31        cout<<data[i]<<" ";
32    }
33
34    cout<<endl;
35    cout<<"-----"=<endl;
36
37    cout<<"Min: "<<data[9]<<endl;
38    cout<<"Max: "<<data[0]<<endl;
39
40    return 0;
41 }
42
43 void BubbleSort(int array1[], int n)
44 {
45     int i, j, tmp;
46
47     for(i = 1; i <= n; i++)
48     {
49         for(j = n; j >= i; j--)
50         {
51             if(array1[j] > array1[j-1])
52             {
53                 tmp = array1[j];
54
55                 array1[j] = array1[j-1];
56
57                 array1[j-1] = tmp;
58             }
59         }
60     }
61 }
```

The terminal output shows the initial state of the array [4, 8, 5, 9, 6, 2, 7, 5, 9, 5], followed by the sorted state [9, 9, 8, 7, 6, 5, 5, 5, 4, 2]. It also displays the minimum value (2) and maximum value (9). The process exits after 0.09283 seconds.

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. Pembandingan alamatnya dapat dimulai dari data yang paling awal atau yang paling akhir. Apabila elemen yang sekarang (sebelumnya) lebih besar dari elemen berikutnya, maka posisinya 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 menyusunkan sedemikian rupa sehingga elemen yang lain lebih kecil dari pada pivot teletak 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
Laporan Akhir - Perkuliahan 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 $i-1$ dengan data yg dicari atau dimaksud. 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 dalam array.

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

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

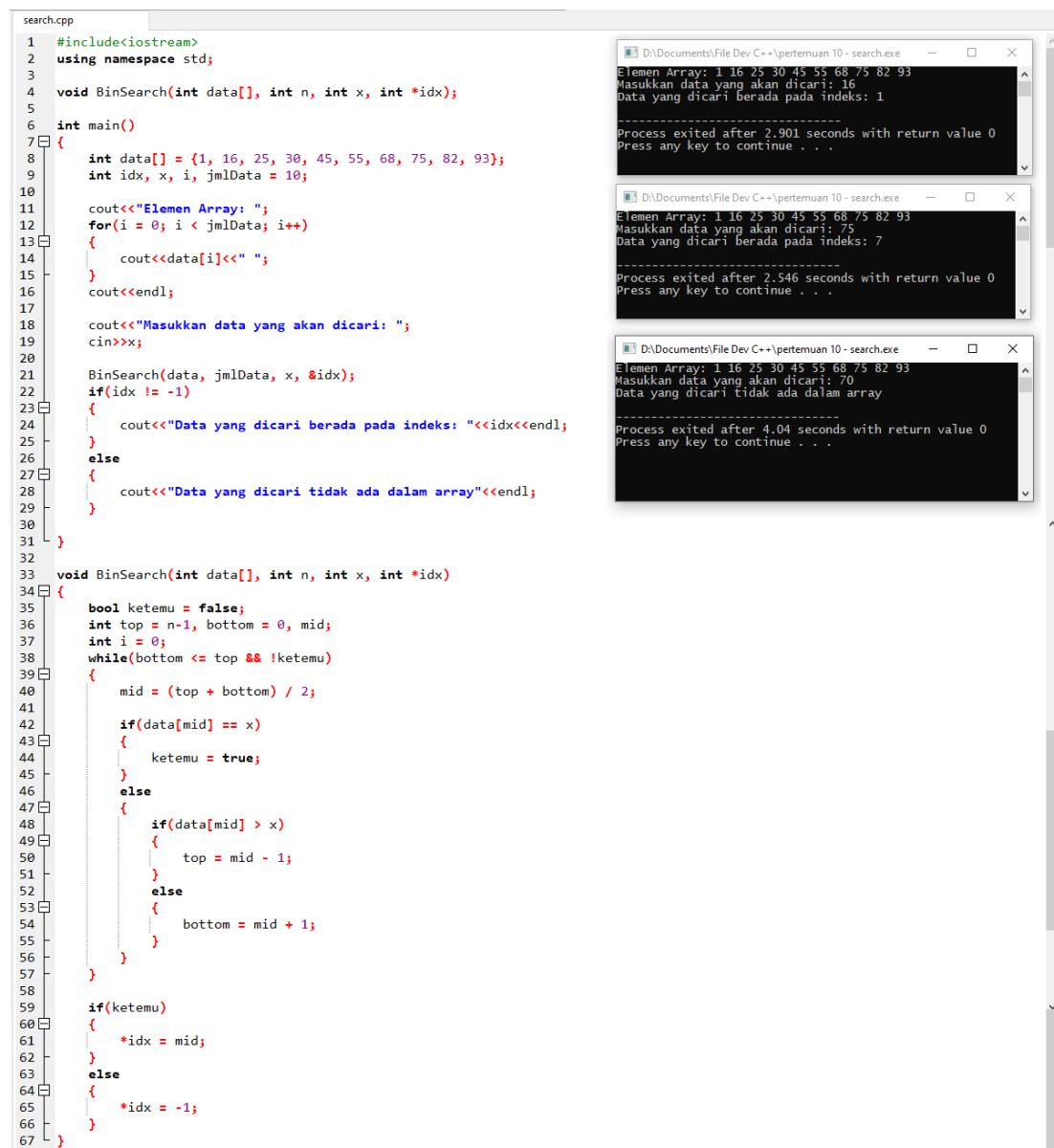
Kekurangan: apabila data yg dicari letaknya pada data terakhir maka dalam penggunaan waktunya, proses ini berjalan lama.

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



4. Buatlah contoh algoritma dan program sederhana menggunakan searching dengan memakai flowchart!

Jawaban No. 4



The image shows a C++ code editor with a file named "search.cpp". The code contains two implementations of a search algorithm: a simple linear search and a binary search.

search.cpp

```
1 #include<iostream>
2 using namespace std;
3
4 void BinSearch(int data[], int n, int x, int *idx);
5
6 int main()
7 {
8     int data[] = {1, 16, 25, 30, 45, 55, 68, 75, 82, 93};
9     int idx, x, i, jmlData = 10;
10
11    cout<<"Elemen Array: ";
12    for(i = 0; i < jmlData; i++)
13    {
14        cout<<data[i]<<" ";
15    }
16    cout<<endl;
17
18    cout<<"Masukkan data yang akan dicari: ";
19    cin>>x;
20
21    BinSearch(data, jmlData, x, &idx);
22    if(idx != -1)
23    {
24        cout<<"Data yang dicari berada pada indeks: "<<idx<<endl;
25    }
26    else
27    {
28        cout<<"Data yang dicari tidak ada dalam array"<<endl;
29    }
30
31 }
32
33 void BinSearch(int data[], int n, int x, int *idx)
34 {
35     bool ketemu = false;
36     int top = n-1, bottom = 0, mid;
37     int i = 0;
38     while(bottom <= top && !ketemu)
39     {
40         mid = (top + bottom) / 2;
41
42         if(data[mid] == x)
43         {
44             ketemu = true;
45         }
46         else
47         {
48             if(data[mid] > x)
49             {
50                 top = mid - 1;
51             }
52             else
53             {
54                 bottom = mid + 1;
55             }
56         }
57
58         if(ketemu)
59         {
60             *idx = mid;
61         }
62         else
63         {
64             *idx = -1;
65         }
66     }
67 }
```

The terminal windows show the following outputs:

- Terminal 1: Elemen Array: 1 16 25 30 45 55 68 75 82 93
Masukkan data yang akan dicari: 16
Data yang dicari berada pada indeks: 1
- Terminal 2: Elemen Array: 1 16 25 30 45 55 68 75 82 93
Masukkan data yang akan dicari: 75
Data yang dicari berada pada indeks: 7
- Terminal 3: Elemen Array: 1 16 25 30 45 55 68 75 82 93
Masukkan data yang akan dicari: 70
Data yang dicari tidak ada dalam array

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, jumlahData = 10;

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

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

    BinSearch(data, jumlahData, 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

The screenshot shows a C++ development environment with two windows. On the left is the code editor window titled "Pertemuan 13 tugas akhir.cpp". It contains the following C++ code:

```
1 #include<iostream>
2 using namespace std;
3
4 int linearSearch(int data[], int dataSize, int element);
5
6 int main()
7 {
8     int dataSize = 5;
9     int data[dataSize];
10    int searchingElement = 0;
11    int flag = 0;
12    cout<<"Masukkan 5 buah data berikut: "<<endl;
13    for(int i = 0; i < dataSize; i++)
14    {
15        cout<<"Masukkan data ke "<<i+1<<": ";
16        cin>>data[i];
17    }
18
19    cout<<"Isi dari array adalah: "<<endl;
20    cout<<"Array: "<<"\t\t" <<"data: "<<endl;
21    for(int count = 0; count < dataSize; count++)
22    {
23        cout<<"array["<<count<<"] "<<"\t\t";
24        cout<<data[count]<<endl;
25    }
26
27    cout<<"\n\nMasukkan data yang dicari: ";
28    cin>>searchingElement;
29
30    flag = linearSearch(data, dataSize, searchingElement);
31
32    if(flag != -1)
33    {
34        cout<<"\n\nData tersebut ditemukan pada posisi: array["<<flag<<"];
35    }
36    else
37    {
38        cout<<"\n\nData tersebut tidak ditemukan"<<endl;
39    }
40
41    return 0;
42 }
43
44 int linearSearch(int data[], int dataSize, int element)
45 {
46     int flag = -1;
47     for(int count = 0; count < dataSize; count++)
48     {
49         if(element == data[count])
50         {
51             flag = count;
52             break;
53         }
54     }
55     return flag;
56 }
```

The terminal window on the right shows the execution of the program. It prompts the user to enter 5 data values, which are then displayed in a tabular format. The user then enters a search value (69) and the program outputs that the value was found at index 0.

```
Masukkan 5 buah data berikut:
Masukkan data ke 1: 69
Masukkan data ke 2: 124
Masukkan data ke 3: 643
Masukkan data ke 4: 22
Masukkan data ke 5: 51
Isi dari array adalah:
Array:           data:
array[0]          69
array[1]          124
array[2]          643
array[3]          22
array[4]          51

Masukkan data yang dicari: 69

Data tersebut ditemukan pada posisi: array[0]
Process exited after 21.27 seconds with return value 0
Press any key to continue . . .
```

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 persatuan 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 ekslusif dalam ilmu komputer. Sebuah pencarian biner mencari nilai tengah (median), melakukan sebuah pembandingan 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.