

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-6



Disusun Oleh:

Nama: Andri Firman Saputra

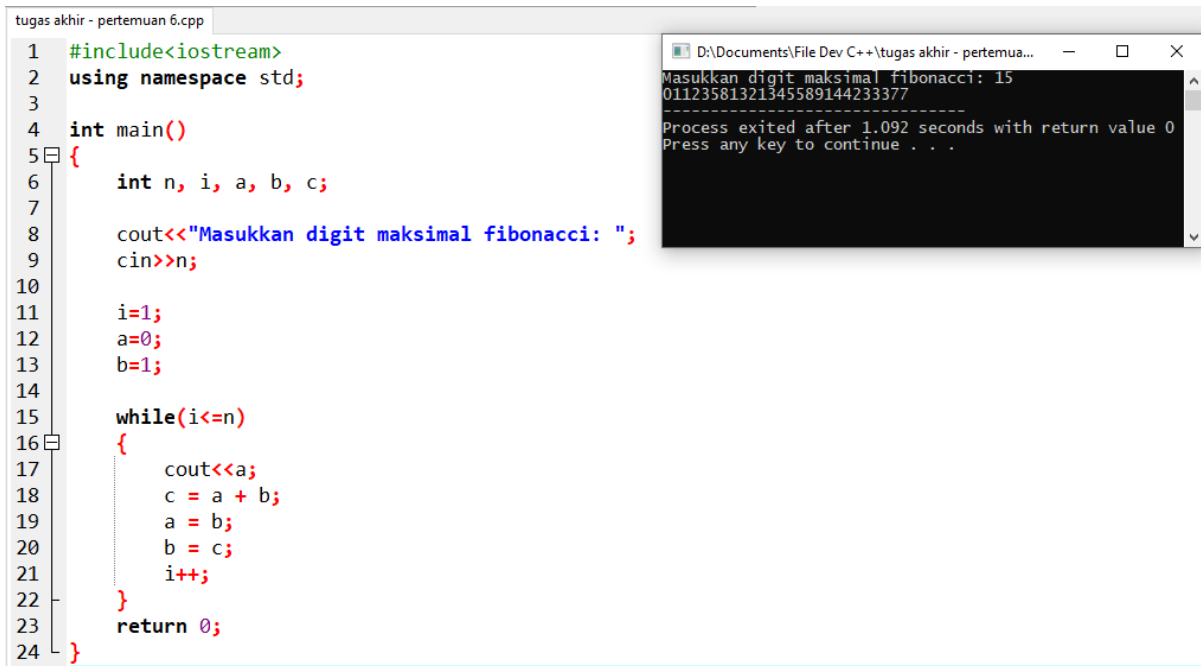
NIM : 201011402125

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 image shows a C++ program in a code editor and its execution output in a console window. The program is a Fibonacci sequence generator. The code editor shows the following 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 console window shows the output of the program. It prompts the user to enter the maximum digit for the Fibonacci sequence. The user enters 15. The program then outputs the Fibonacci sequence: 01123581321345589144233377. The console window also shows the message "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 baru)

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:

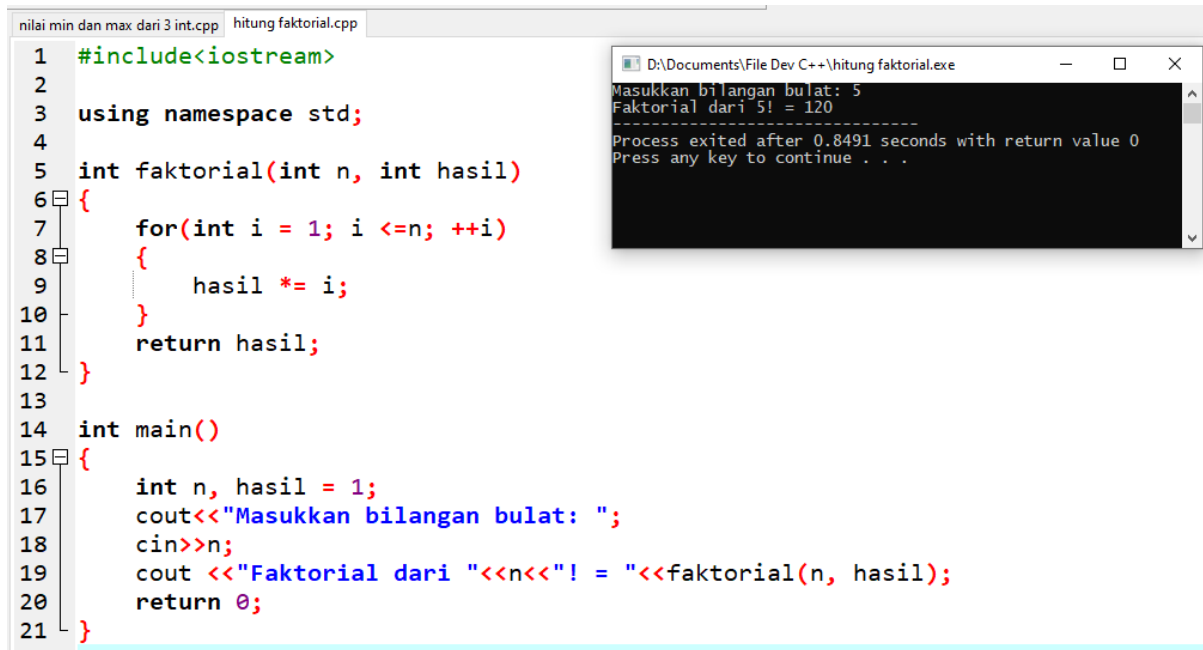
nama_fungsi (parameter jika ada);

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 dikeluarkan / diemballkan. 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 memaui flowchart!

Jawaban No. 4

Contoh Program



The image shows a screenshot of a C++ program in a code editor and its execution output in a console window. The code editor has two tabs: 'nilai min dan max dari 3 int.cpp' and 'hitung faktorial.cpp'. The 'hitung faktorial.cpp' tab is active, showing the following 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 }
```

The console window, titled 'D:\Documents\File Dev C++\hitung faktorial.exe', shows the program's 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:

