**LAPORAN HASIL ANALISA TUGAS PRAKTIKUM KE-6**

**(BAGIAN C, D, DAN E)**

**MATA KULIAH PEMROGRAMAN DASAR**



**OLEH :**

**REFILA DYAH GHIZANDA WARDOYO**

**21091397041**

**D4 MANAJEMEN INFORMATIKA**

**UNIVERSITAS NEGERI SURABAYA**

**2021/2022**

**C. TUGAS PENDAHULUAN**

1. Rekursi adalah suatu proses yang bisa memanggil dirinya sendiri. Dalam sebuah rekursi, terdapat pengertian sebuah prosedur atau fungsi. Perbedaannya adalah rekursi bisa memanggil dirinya sendiri, kalau prosedur atau fungsi harus dipanggil melalui pemanggil prosedur atau fungsi.
2. Fungsi untuk menghitung nilai faktorial :

n!= n . (n-1)! Untuk n>1

0!=1 Untuk n=0 atau n=1

Secara pemrograman dapat ditulis sebagai berikut :

Faktorial (0) = 1 (1)

Faktorial (N) = N\*Faktorial( N – 1) (2)

Persamaan (1) tidak bersifat rekursif, disebut nilai awal atau basis. Persamaan (2) contoh hubungan rekurens yang artinya nilai suatu fungsi dengan argumen tertentu bisa dihitung dari fungsi yang sama dengan argumen yang lebih kecil.

1. Fungsi untuk menampilkan nilai fibbonacci dari deret fibbonacci :

Fibbonacci(1) = 1 (1)

Fibbonacci(2) = 2 (2)

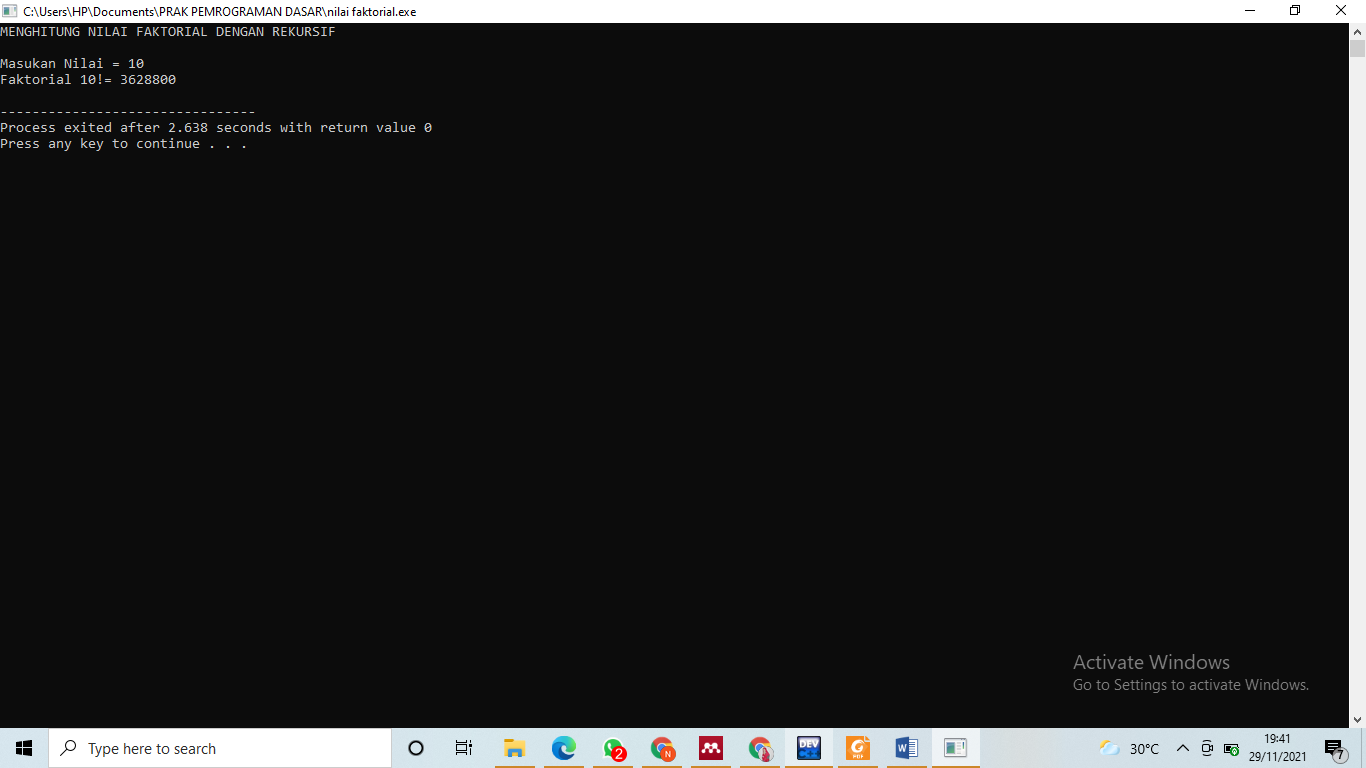
Fibbonacci(N) = Fibbonacci(N-1) + Fibbonacci(N-2) (3)

**D. PERCOBAAN**

Percobaan 1: Fungsi rekursif untuk menghitung nilai faktorial

#include <iostream>  
using namespace std;  
long int faktorial (int A);  
int main(){  
int r,hasil;  
cout<<"MENGHITUNG NILAI FAKTORIAL DENGAN REKURSIF"<<endl;  
cout<<endl;  
cout<<"Masukan Nilai = ";  
cin>>r;  
hasil=faktorial(r);  
cout<<"Faktorial "<<r<<"!= "<<hasil<<endl;  
}  
long int faktorial (int A){  
if (A==1)  
return(A);  
else  
return (A\*faktorial(A-1));  
}

Hasil percobaan :

****

Hasil analisa :

#include <iostream>  
using namespace std;

Merupakan pemberian intruksi untuk preprocessor directive #include agar menyertakan file header **iostream** sehingga memungkinkan penggunaan perintah cin, cout, dan endl dalam program ini. *Using namespace std;* berkaitan dengan fungsi dari file header iostream karena berisi tentang perintah seperti cout, cin, dan endl. Using namespace std; merupakan pernyataan (statement) yang selalu diakhiri dengan tanda ; (titik koma).

long int faktorial (int A);

Merupakan pendeklarasian fungsi faktorial dengan menggunakan parameter formal fungsi tipe data int variabel A. Bentuk umum dari pendeklarasian fungsi seperti tipe\_fungsi nama\_fungsi (parameter\_fungsi);

int main(){

Merupakan fungsi main yang berisi statement-statement yang selalu diawali dengan tanda { (kurung kurawal buka) dan dikhiri dengan tanda } (kurung kurawal tutup). Fungsi main akan menjalankan statement-statement yang berada di dalamnya secara berurutan. Tanda { (kurung kurawal buka) merupakan awal dari fungsi main.

int r,hasil;

Merupakan pendeklarasian variabel r dan hasil dengan tipe data integer.

cout<<"MENGHITUNG NILAI FAKTORIAL DENGAN REKURSIF"<<endl;

Merupakan perintah untuk menampilkan teks MENGHITUNG NILAI FAKTORIAL DENGAN REKURSIF dilanjutkan dengan perintah untuk menapilkan teks berikutnya di baris baru.

cout<<"Masukan Nilai = ";

Merupakan perintah untuk menampilkan teks Masukan Nilai.

cin>>r;

.

Merupakan perintah untuk menginputkan nilai variable r.

hasil=faktorial(r);

Merupakan pendefinisian variabel hasil yang merujuk pada fungsi factorial variabel r.

cout<<"Faktorial "<<r<<"!= "<<hasil<<endl;

Merupakan perintah untuk menampilkan output dalam bentuk kalimat Faktorial dengan menampilkan variabel r yang telah diinputkan oleh keyboard dan menampilkan tanda != beserta dengan variabel hasil pada hasil program. Kemudian berpindah pada baris baru. Tanda } (kurung kurawal tutup) merupakan akhir dari fungsi main.

long int faktorial (int A){

Merupakan fungsi long int factorial dengan menggunakan parameter formal fungsi tipe data int variabel A yang berisi statement - statement yang selalu diawali dengan tanda { (kurung kurawal buka) dan dikhiri dengan tanda } (kurung kurawal tutup). Fungsi long int faktorial akan menjalankan statement-statement yang berada di dalamnya secara berurutan. Tanda { (kurung kurawal buka) merupakan awal dari fungsi long int faktorial.

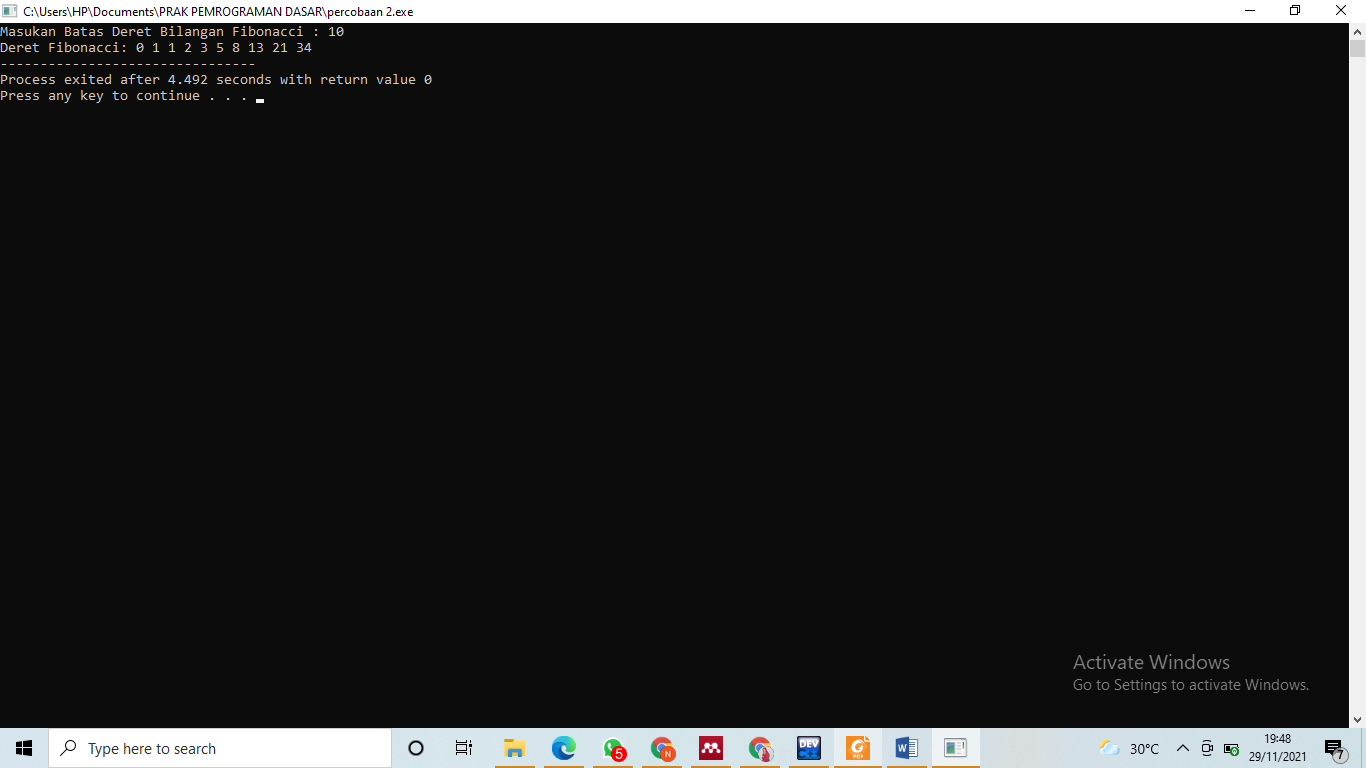
if (A==1)  
return(A);  
else  
return (A\*faktorial(A-1));  
}

Merupakan kondisi jika nilai yang dimasukkan adalah 1 maka nilai balik (return value) adalah nilai itu sendiri. Else return (A\*factorial(A-1)); kondisi jika tidak (selain) maka nilai balik yang dimasukkan akan dihitung menggunakan rumus factorial tersebut. Tanda } (kurung kurawal tutup) merupakan akhir dari fungsi long int faktorial.

Percobaan 2: Fungsi untuk menampilkan deret fibonaci

#include <iostream>  
using namespace std;  
int fibonacci(int m) {  
if (m == 0 || m ==1)  
{  
return m;  
}  
else  
{  
return (fibonacci(m-1) + fibonacci(m-2));  
}  
}  
int main() {  
int n, m= 0;  
cout << "Masukan Batas Deret Bilangan Fibonacci : ";  
cin >> n;  
cout << "Deret Fibonacci: ";  
for (int i = 1; i <= n; i++){  
cout << fibonacci(m) <<" ";  
m++;  
}  
return 0;  
}

Hasil percobaan :

****

Hasil analisa :

#include <iostream>  
using namespace std;

Merupakan pemberian intruksi untuk preprocessor directive #include agar menyertakan file header **iostream** sehingga memungkinkan penggunaan perintah cin, cout, dan endl dalam program ini. *Using namespace std;* berkaitan dengan fungsi dari file header iostream karena berisi tentang perintah seperti cout, cin, dan endl. Using namespace std; merupakan pernyataan (statement) yang selalu diakhiri dengan tanda ; (titik koma).

int fibonacci(int m){

Merupakan pendeklarasian fungsi Fibonacci dengan menggunakan parameter formal fungsi tipe data integer variable m. Bentuk umum dari pendeklarasian fungsi seperti tipe\_fungsi nama\_fungsi (parameter\_fungsi); Tanda { (kurung kurawal buka) merupakan awal dari fungsi int fibonacci.

if (m == 0 || m ==1)  
{  
return m;  
}  
else  
{  
return (fibonacci(m-1) + fibonacci(m-2));  
}  
}

Kondisi jika nilai variabel m yang dimasukkan adalah 0 atau 1 maka nilai balik (return value) adalah nilai itu sendiri. }Else{ return (fibonacci(m-1) + fibonacci(m-2)); kondisi tidak (else) maka nilai balik yang dimasukkan akan dihitung menggunakan rumus fibonacci tersebut. Tanda } (kurung kurawal tutup) merupakan akhir dari fungsi int fibonacci.

int main() {

Merupakan fungsi main yang berisi statement-statement yang selalu diawali dengan tanda { (kurung kurawal buka) dan dikhiri dengan tanda } (kurung kurawal tutup). Fungsi main akan menjalankan statement-statement yang berada di dalamnya secara berurutan. Tanda { (kurung kurawal buka) merupakan awal dari fungsi main.

int n, m= 0;

Merupakan pendeklarasian variabel n dan m yang bernilai 0 dengan tipe data integer.

cout << "Masukan Batas Deret Bilangan Fibonacci : ";

Merupakan perintah untuk menampilkan teks Masukan Batas Deret Bilangan Fibonacci :

Merupakan perintah untuk menginputkan nilai variable n.

cin >> n;

Merupakan perintah untuk menginputkan nilai variable n.

Merupakan perintah untuk menampilkan teks Deret Fibonacci:

cout << "Deret Fibonacci: ";

for (int i = 1; i <= n; i++){

Merupakan perintah untuk memberi nilai awal perulangan yang dimulai dari i=1. Pendeklarasian variabel i tipe data int digunakan untuk perulangan. Perulangan akan berhenti jika nilai i <= (lebih besar sama dengan) inputan nilai n. *i++* perulangan bersifat increment atau akan ditambah 1 setiap kali proses hingga sampai batas yang ditentukan maka proses akan berhenti. Tanda { (kurung kurawal buka) merupakan awal dari fungsi for.

cout << fibonacci(m) <<" ";  
m++;  
}

Merupakan perindah menampilkan output fungsi Fibonacci variabel m dan memberikan spasi. *M++* perulangan bersifat increment atau akan ditambah 1 setiap kali proses hingga sampai batas yang ditentukan maka proses akan berhenti.Tanda } (kurung kurawal tutup) merupakan akhir dari fungsi for.

return 0;

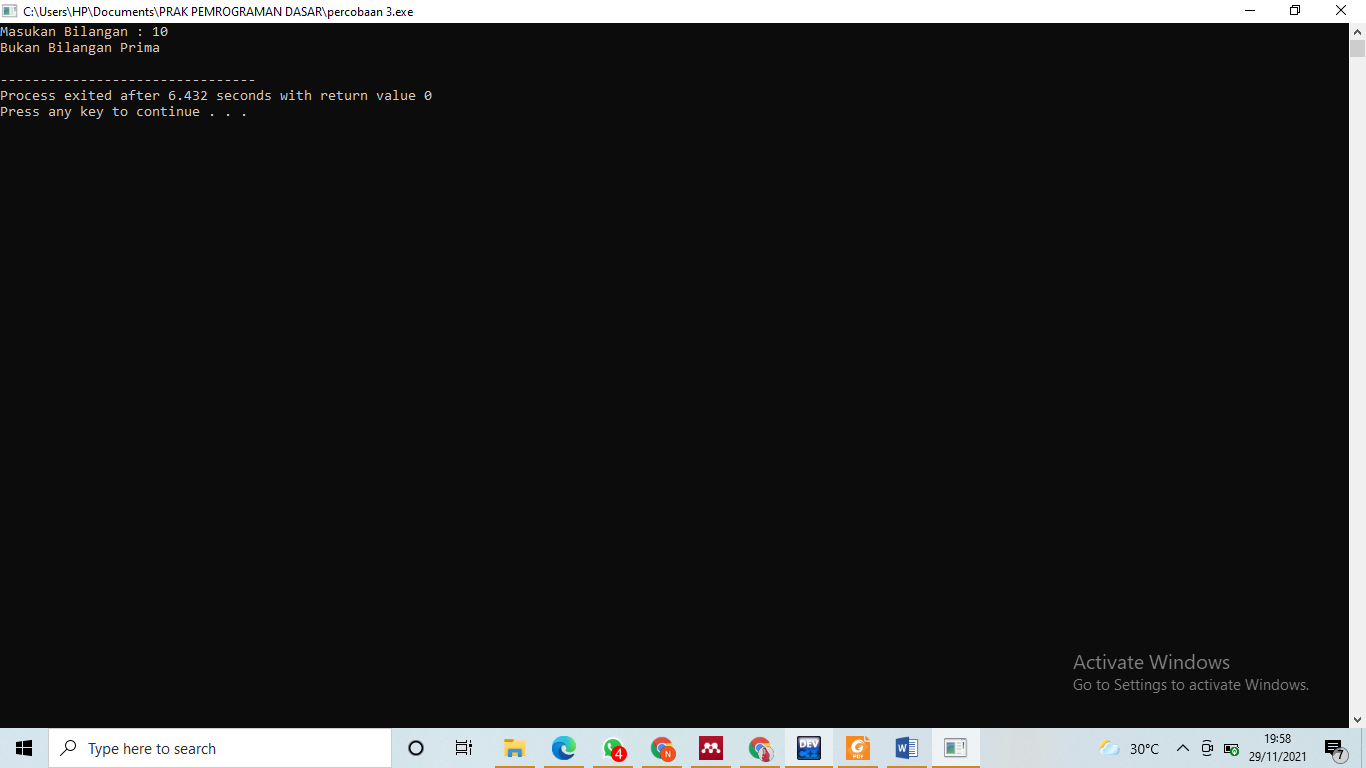
}

Menyatakan hasil keluaran fungsi main adalah 0 atau untuk memberikan exit status yang berarti program berakhir dengan normal. Tanda } (kurung kurawal tutup) merupakan akhir dari fungsi main.

Percobaan 3: Fungsi rekursi untuk menentukan bilangan prima atau bukan prima

#include <iostream>  
using namespace std;  
int ambil(int bil, int i){  
if (i == 1) {  
return 1;  
}  
else if (bil % i == 0) {  
return 1 + ambil(bil, --i);  
} else {  
return 0 + ambil(bil, --i);  
}  
}  
int cek(int bil){  
if (bil > 1) {  
return (ambil(bil, bil) == 2);  
} else  
return false;  
}  
int main(){  
int bil;  
cout<<"Masukan Bilangan : ";  
cin>>bil;  
if (cek(bil)){  
cout<<"Bilangan Prima"<<endl;  
}else {  
cout<<"Bukan Bilangan Prima"<<endl;  
}  
return 0;  
}

Hasil percobaan :

****

Hasil analisa :

#include <iostream>  
using namespace std;

Merupakan pemberian intruksi untuk preprocessor directive #include agar menyertakan file header **iostream** sehingga memungkinkan penggunaan perintah cin, cout, dan endl dalam program ini. *Using namespace std;* berkaitan dengan fungsi dari file header iostream karena berisi tentang perintah seperti cout, cin, dan endl. Using namespace std; merupakan pernyataan (statement) yang selalu diakhiri dengan tanda ; (titik koma).

int ambil(int bil, int i){

Merupakan pendeklarasian variabel ambil dengan tipe data integer variabel bil dan i. Bentuk umum dari pendeklarasian fungsi seperti tipe\_fungsi nama\_fungsi (parameter\_fungsi); Tanda { (kurung kurawal buka) merupakan awal dari fungsi int ambil.

if (i == 1) {  
return 1;  
}  
else if (bil % i == 0) {  
return 1 + ambil(bil, --i);  
} else {  
return 0 + ambil(bil, --i);  
}  
}

Merupakan kondisi jika nilai variabel i yang dimasukkan adalah 1 maka nilai balik (return value) adalah nilai 1. *}Else if (bil%i==0){ return 1+ambil(bil, --i);* kondisi jika tidak (else if) apabila nilai variabel bil modulo i = 0 maka nilai balik yang dimasukkan akan dihitung menggunakan rumus bilangan prima tersebut. *}Else { return 0+ambil(bil, --i);* kondisi jika tidak (else) maka nilai balik yang dimasukkan akan dihitung menggunakan rumus bilangan prima tersebut. Tanda } (kurung kurawal tutup) merupakan akhir dari fungsi int ambil.

int cek(int bil){

Merupakan pendeklarasian variabel cek dengan tipe data integer variabel bil. Bentuk umum dari pendeklarasian fungsi seperti tipe\_fungsi nama\_fungsi (parameter\_fungsi); Tanda { (kurung kurawal buka) merupakan awal dari fungsi int cek.

if (bil > 1) {  
return (ambil(bil, bil) == 2);  
} else  
return false;  
}

Merupakan kondisi jika nilai variabel bil yang dimasukkan adalah (lebih dari) >1 atau bernilai benar maka nilai balik (return value) adalah nilai 2. *}Else return false;* kondisi tidak (else) maka nilai balik yang dimasukkan akan bernilai salah. Tanda } (kurung kurawal tutup) merupakan akhir dari fungsi int cek.

int main(){

Merupakan fungsi main yang berisi statement-statement yang selalu diawali dengan tanda { (kurung kurawal buka) dan dikhiri dengan tanda } (kurung kurawal tutup). Fungsi main akan menjalankan statement-statement yang berada di dalamnya secara berurutan. Tanda { (kurung kurawal buka) merupakan awal dari fungsi main.

int bil;

Merupakan pendeklarasian variabel bil dengan tipe data integer.

cout<<"Masukan Bilangan : ";

Merupakan perintah untuk menampilkan teks Masukan Bilangan.

cin>>bil;

Merupakan perintah untuk menginputkan nilai variable bil.

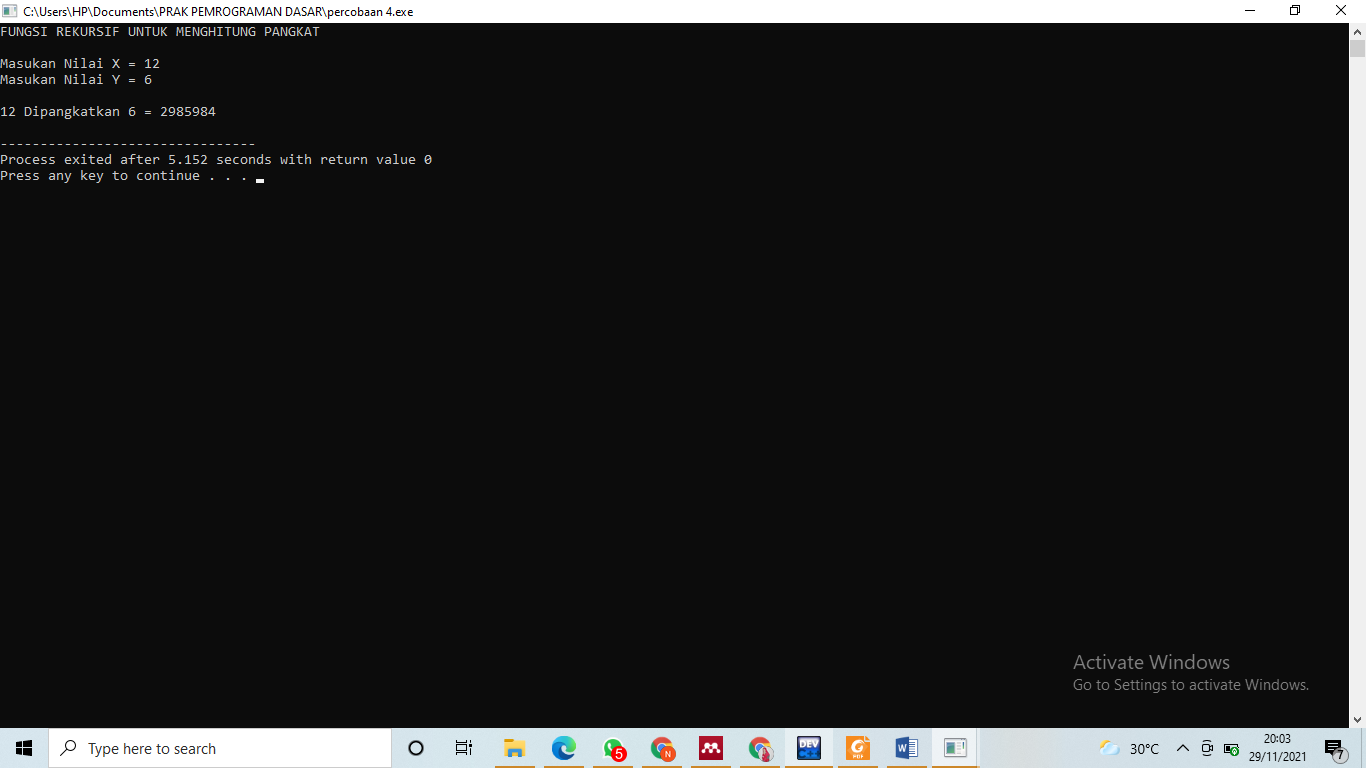
if (cek(bil)){  
cout<<"Bilangan Prima"<<endl;  
}else {  
cout<<"Bukan Bilangan Prima"<<endl;  
}  
return 0;  
}

Merupakan kondisi jika nilai yang dimasukkan bernilai benar di cek bil maka akan menampilkan output dalam bentuk kalimat Bilangan Prima pada hasil program.}else { *cout<<”Bukan Bilangan Prima”<<endl;}* kondisi tidak (else) nilai yang dimasukkan salah maka akan menampilkan menampilkan output dalam bentuk kalimat Bukan Bilangan Prima pada hasil program.

Percobaan 4: Fungsi rekursi untuk menghitung pangkat

#include <iostream>  
using namespace std;  
long int pangkatrekursif(int x, int y);  
int main(){  
int x,y;  
cout<<"FUNGSI REKURSIF UNTUK MENGHITUNG PANGKAT"<<endl;  
cout<<endl;  
cout<<"Masukan Nilai X = ";  
cin>>x; cout<<"Masukan Nilai Y = ";  
cin>>y;  
cout<<endl;  
cout<<x<<" Dipangkatkan "<<y<<" = "<<pangkatrekursif(x,y)<<endl;  
}  
long int pangkatrekursif(int x, int y){  
if (y==0)  
return 1 ;  
else  
return x \* pangkatrekursif(x,y-1); }

Hasil percobaan :

****

Hasil analisa :

#include <iostream>  
using namespace std

Merupakan pemberian intruksi untuk preprocessor directive #include agar menyertakan file header **iostream** sehingga memungkinkan penggunaan perintah cin, cout, dan endl dalam program ini. *Using namespace std;* berkaitan dengan fungsi dari file header iostream karena berisi tentang perintah seperti cout, cin, dan endl. Using namespace std; merupakan pernyataan (statement) yang selalu diakhiri dengan tanda ; (titik koma).

long int pangkatrekursif(int x, int y);

Merupakan fungsi long integer pangkatrekursif dengan menggunakan parameter formal fungsi tipe data integer variabel x dan y. Bentuk umum dari pendeklarasian fungsi seperti tipe\_fungsi nama\_fungsi (parameter\_fungsi);

int main(){

Merupakan fungsi main yang berisi statement-statement yang selalu diawali dengan tanda { (kurung kurawal buka) dan dikhiri dengan tanda } (kurung kurawal tutup). Fungsi main akan menjalankan statement-statement yang berada di dalamnya secara berurutan. Tanda { (kurung kurawal buka) merupakan awal dari fungsi main.

int x,y;

Merupakan pendeklarasian variabel x dan y dengan tipe data integer.

cout<<"FUNGSI REKURSIF UNTUK MENGHITUNG PANGKAT"<<endl;

Merupakan perintah untuk menampilkan teks dilanjutkan dengan perintah untuk menapilkan teks FUNGSI REKURSIF UNTUK MENGHITUNG PANGKAT berikutnya di baris baru.

cout<<"Masukan Nilai X = ";

Merupakan perintah untuk menampilkan teks Masukan Nilai X =

cin>>x;

Merupakan perintah untuk menginputkan nilai variable x.

cout<<"Masukan Nilai Y = ";

Merupakan perintah untuk menampilkan teks Masukan Nilai Y =

cin>>y;

Merupakan perintah untuk menginputkan nilai variable y.

cout<<x<<" Dipangkatkan "<<y<<" = "<<pangkatrekursif(x,y)<<endl;  
}

Merupakan perintah untuk menampilkan menampilkan output nilai variabel x dengan kalimat Dipangkatkan nilai variabel y yang sama-sama diinputkan oleh keyboard dan menampilkan tanda = beserta dengan variabel pangkatrekursif(x,y) pada hasil program. Kemudian berpindah pada baris baru. Tanda } (kurung kurawal tutup) merupakan akhir dari fungsi main.

long int pangkatrekursif(int x, int y)

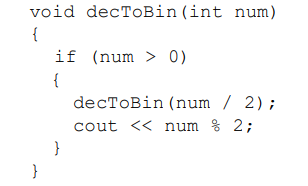
Merupakan fungsi long integer pangkatrekursif dengan menggunakan parameter formal fungsi tipe data integer variabel x dan y yang berisi statement-statement yang selalu diawali dengan tanda { (kurung kurawal buka) dan dikhiri dengan tanda } (kurung kurawal tutup). Fungsi long int pangkat rekursif akan menjalankan statement-statement yang berada di dalamnya secara berurutan. Tanda { (kurung kurawal buka) merupakan awal dari fungsi long int pangkat rekursif.

{  
if (y==0)  
return 1 ;  
else  
return x \* pangkatrekursif(x,y-1); }

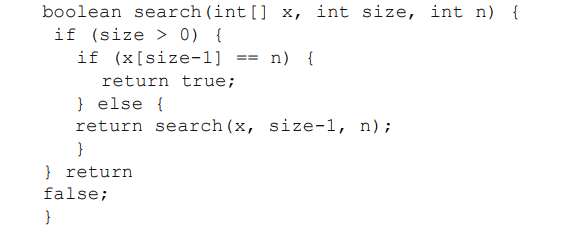
Merupakan kondisi jika nilai variabel y yang dimasukkan adalah 1 maka nilai balik (return value) adalah nilai 1. *Else return x \* pangkatrekursif (x,y-1);* kondisi tidak (else) maka nilai balik yang dimasukkan akan dihitung menggunakan rumus pangkat tersebut. Tanda } (kurung kurawal tutup) merupakan akhir dari fungsi long int pangkat rekursif.

**E. LATIHAN**

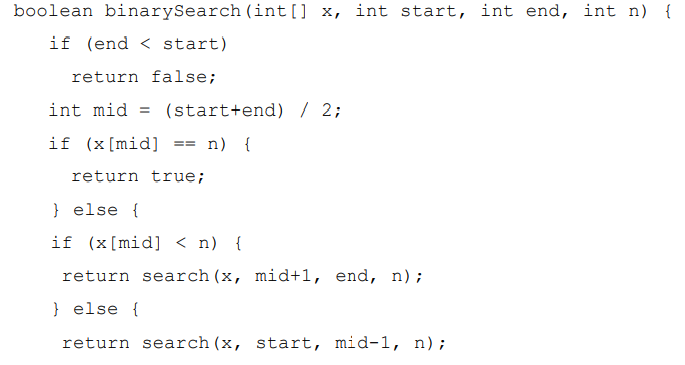
1. Buatlah program rekursif untuk menghitung segitiga pascal!
2. Buatlah program secara rekursif, masukkan jumlah N karakter dan cetak dalam semua kombinasi!
3. Buat program binary search dengan rekursif! (data tentukan sendiri)
4. Buatlah program rekursif untuk memecahkan permasalahan Menara Hanoi!
5. Jelaskan proses rekursif untuk program dibawah ini!



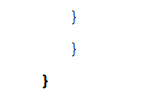
1. Jelaskan proses rekursif untuk program dibawah ini!



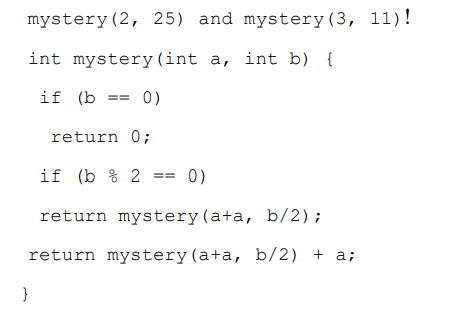
1. Jelaskan proses rekursif untuk program dibawah ini!



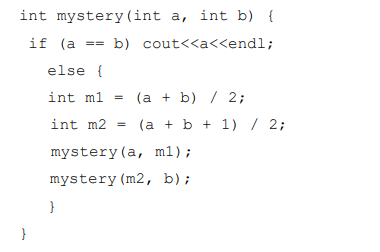




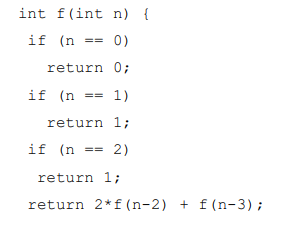
1. Jelaskan proses rekursif untuk program dibawah ini dengan memanggil mystery(2, 25) and mystery(3, 11)!



1. Jelaskan proses rekursif untuk program dibawah ini dengan memanggil mystery(0, 8)!



1. . Jelaskan proses rekursif untuk program dibawah ini!



1. Jelaskan proses rekursif untuk program dibawah ini dengan memanggil square(5), cube(5), cube(123)?

