**C++ 9 Code.**

**RECURSIVE FUNCTION**

CODE 1:

#include <iostream>

using namespace std;

int factorial(int n) {

if (n <= 1)

return 1;

else

return n \* factorial(n - 1);

}

int main() {

int n;

cout<<"Enter A Number To Find Its FacTorial:";

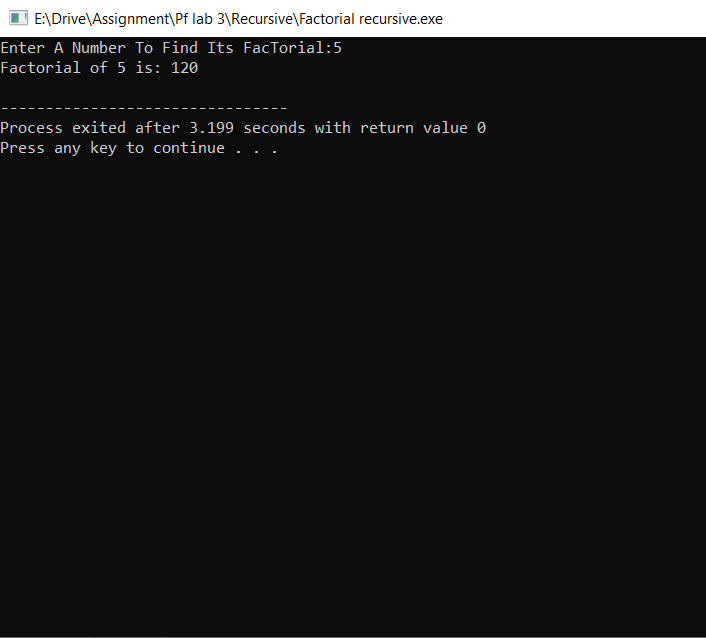
cin>>n;

cout << "Factorial of "<<n<<" is: " << factorial(n) << std::endl;

return 0;

}

OUTPUT:



Code 2:

#include <iostream>

using namespace std;

int fibonacci(int n) {

if (n <= 1)

return n;

else

return fibonacci(n - 1) + fibonacci(n - 2);

}

int main() {

int n;

cout<<"Enter A Number To Print the Febonincial series:";

cin>>n;

for (int i = 0; i < n; ++i) {

std::cout << fibonacci(i) << " ";

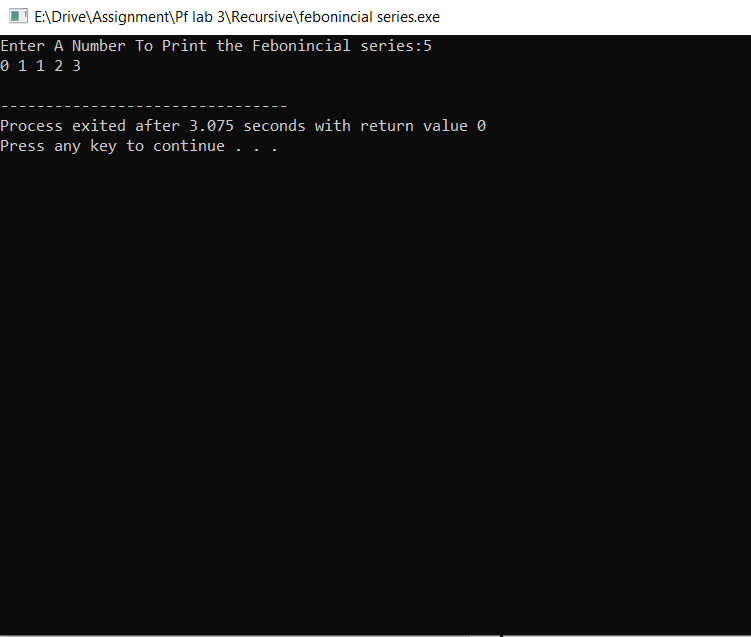
}

cout << endl;

return 0;

}

OutPut:



Code 3:

#include <iostream>

using namespace std;

int sum(int n) {

if (n == 0)

return 0;

else

return n + sum(n - 1);

}

int main() {

int n;

cout<<"Enter Length Of Natural Number TO FIND ITS SUM: ";

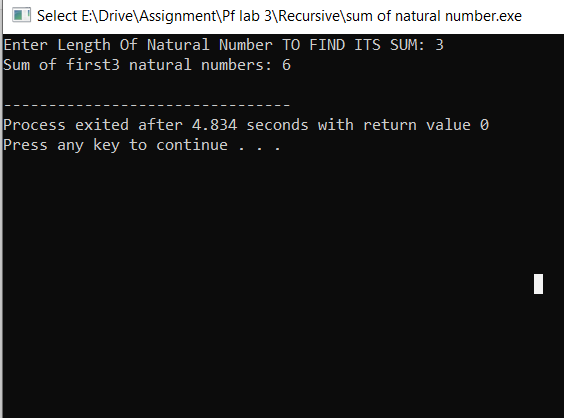
cin>>n;

cout << "Sum of first" <<n<<" natural numbers: " << sum(n) << std::endl;

return 0;

}

Output:



Code 4:

#include <iostream>

using namespace std;

double power(double base, int exponent) {

if (exponent == 0)

return 1;

else

return base \* power(base, exponent - 1);

}

int main() {

int b,p;

cout<<"Enter Base:";

cin>>b;

cout<<"Enter Power:";

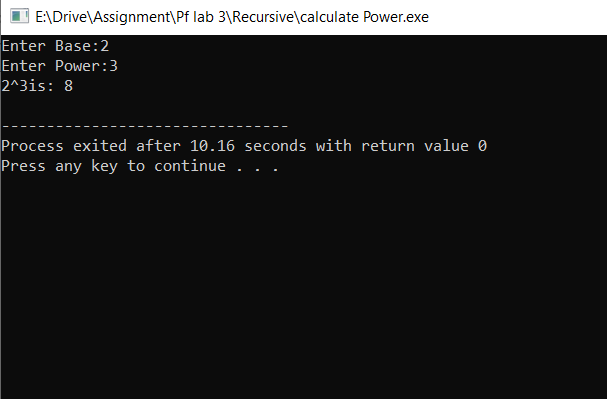
cin>>p;

cout << b<<"^" <<p<<" is: " << power(b, p) << endl;

return 0;

}

Output:



Code 5:

#include <iostream>

using namespace std;

int gcd(int a, int b) {

if (b == 0)

return a;

else

return gcd(b, a % b);

}

int main() {

int a,b;

cout<<"ENTER FIRST AND SECOND NUMBER"<<endl;

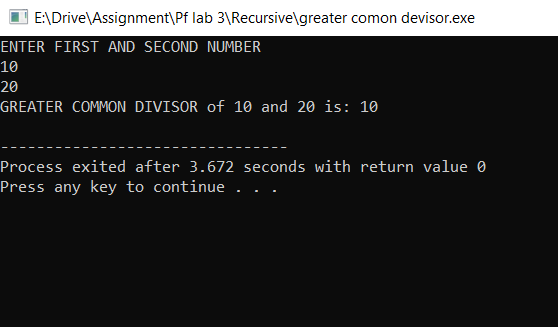
cin>>a>>b;

cout << "GREATER COMMON DIVISOR of "<<a<< " and "<<b<< " is: " << gcd(a, b) << std::endl;

return 0;

}

OutPut:



Code 6:

#include<iostream>

using namespace std;

int ispalodrom(int x,int r);

int main()

{

int num,rev=0;

cout<<"Enter number";

cin>>num;

if(ispalodrom(num,rev))

{

cout<<"it is palindrom";

}

else

{

cout<<"not palindrom";

}

return 0;

}

int ispalodrom(int x,int r)

{

if(x==0)

{

return r;

}

else

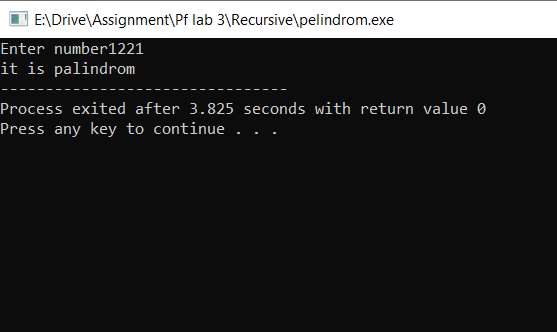
{

r=r\*10+x%10;

return ispalodrom(x/10,r);

}

}

OutPut: 

Code 7:

#include <iostream>

using namespace std;

int binaryToDecimal(int binary, int power) {

if (binary == 0)

return 0;

else

return (binary % 10) \* (1 << power) + binaryToDecimal(binary / 10, power + 1);

}

int main() {

int n;

cout<<"Enter Binary Number:";

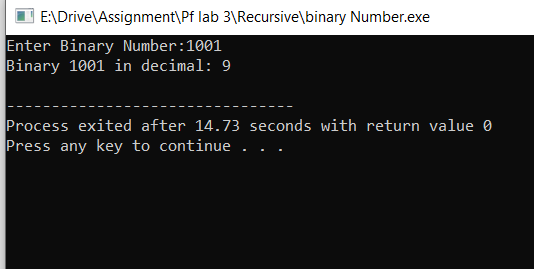
cin>>n;

cout << "Binary "<<n<<" in decimal: " << binaryToDecimal(n, 0) << std::endl;

return 0;

}

OutPut:



Code 8:

#include <iostream>

using namespace std;

//\*\*\*\*\*\*\*

//\*\*\*\*\*\*

//\*\*\*\*

//\*\*\*

//\*\*

//\*

void printPattern(int n) {

if (n <= 0)

return;

else {

for (int i = 0; i < n; ++i)

cout << "\* ";

cout << endl;

printPattern(n - 1);

}

}

int main() {

int n;

cout<<"Enter HOw Many Number Of Staric Print in 1 Line TO Print Pattern :";

cin>>n;

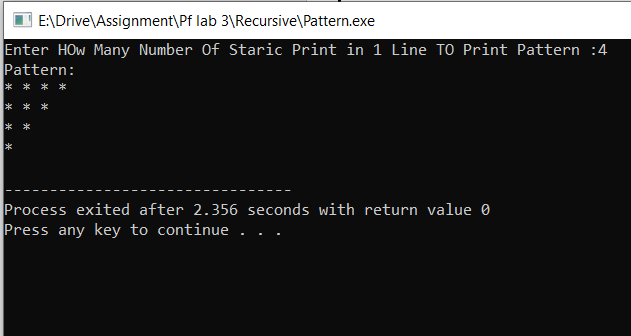
cout << "Pattern:\n";

printPattern(n);

return 0;

}

OutPut:



Code 9:

#include <iostream>

using namespace std;

int countDigits(int num) {

if (num == 0)

return 0;

else

return 1 + countDigits(num / 10);

}

int main() {

int n;

cout<<"Enter A Number:";

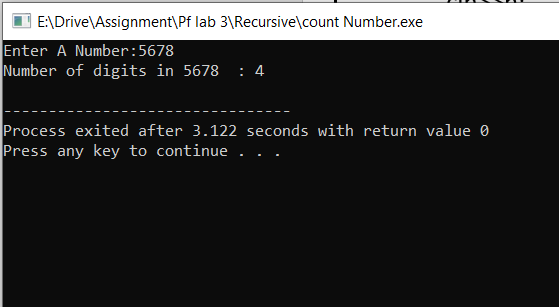
cin>>n;

cout << "Number of digits in "<<n<<" : " << countDigits(n) << std::endl;

return 0;

}

OutPut:



Code 10 :

#include <iostream>

using namespace std;

bool isPrime(int num, int i) {

if (i == 1)

return true;

else if (num % i == 0)

return false;

else

return isPrime(num, i - 1);

}

void printPrimesInRange(int start, int end) {

if (start > end)

return;

else {

if (isPrime(start, start / 2))

cout << start << " ";

printPrimesInRange(start + 1, end);

}

}

int main() {

int f,en;

cout<<"ENTER Start Number:";

cin>>f;

cout<<"Enter End Number:";

cin>>en;

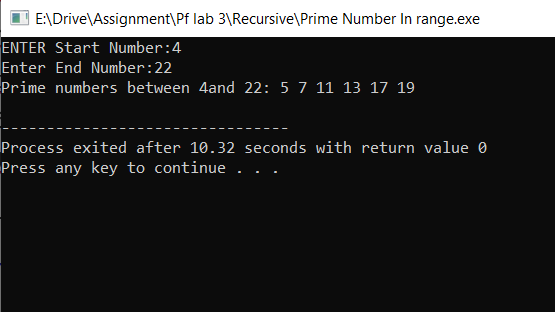
cout << "Prime numbers between "<<f<< "and "<<en<<": ";

printPrimesInRange(f, en);

cout << endl;

return 0;}

Output:



**FUNCTION OVERLOADING**

CODE 1:

#include<iostream>

int add(int a,int b)

{

return a+b;

}

int add(int a,int b,int c)

{

return a+b+c;

}

double add(double a,double b)

{

return a+b;

}

float add(float a,float b)

{

return a+b;

}

float add(float a,float b,float c)

{

return a+b+c;

}

using namespace std;

int main()

{

int a=90 ,b=67, c=90;

float x=5.5,y=5.4,z=10.0;

double j=7837294872384,k=7387492334;

cout<<"sum of two integer values :"<<add(a,b)<<endl;

cout<<"sum of three integer values :"<<add(a,b,c)<<endl;

cout<<"sum of two floating point values :"<<add(x,y)<<endl;

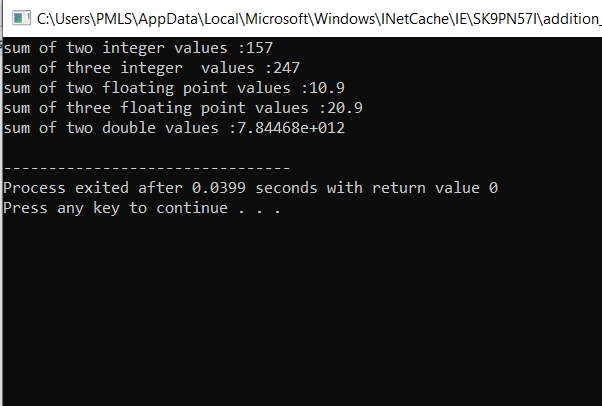
cout<<"sum of three floating point values :"<<add(x,y,z)<<endl;

cout<<"sum of two double values :"<<add(j,k)<<endl;

return 0;

}

OutPut:



Code 2:

#include<iostream>

using namespace std;

float Area(float);

float Area(float,float);

int main()

{

float r,w,l;

cout<<"Radius of Circle:";

cin>>r;

cout<<"Area of Circle"<<Area(r)<<endl;

cout<<"Enter width of Rectangle:"<<endl;

cin>>w;

cout<<"Enter length of Rectangle:"<<endl;

cin>>l;

cout<<"Area of Rectangle:"<<Area(w,l)<<endl;

return 0;

}

float Area(float x)

{

return 3.14\*x\*x;

}

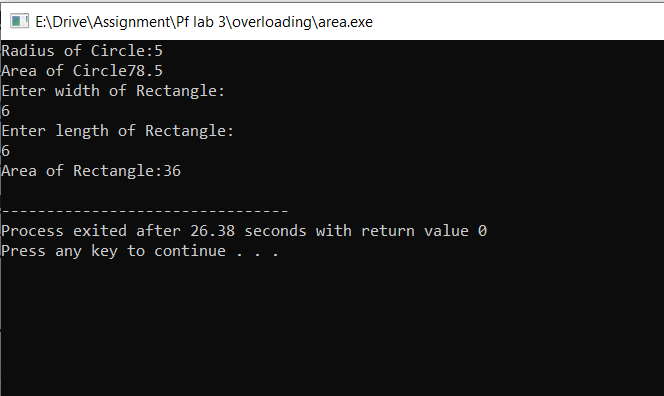
float Area(float x, float y)

{

return x\*y;

}

OutPut:



Code 3:

#include<iostream>

using namespace std;

int mul(int,int);

float mul(float,int);

int mul(int a,int b)

{

return a\*b;

}

float mul(double x, int y)

{

return x\*y;

}

int main()

{

int r1 = mul(6,7);

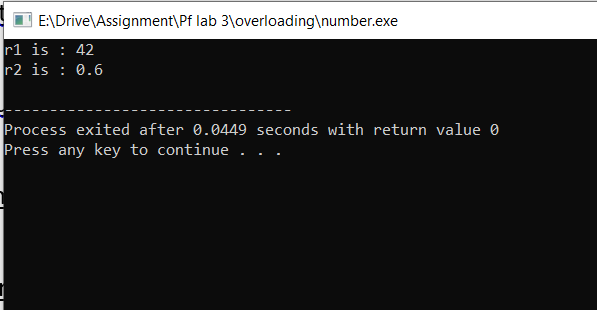
float r2 = mul(0.2,3);

std::cout << "r1 is : " <<r1<< std::endl;

std::cout <<"r2 is : " <<r2<< std::endl;

return 0;

} OutPut



Code 4:

#include <iostream>

using namespace std;

void print(int num) {

cout << "Integer: " << num << endl;

}

void print(double num) {

cout << "Double: " << num << endl;

}

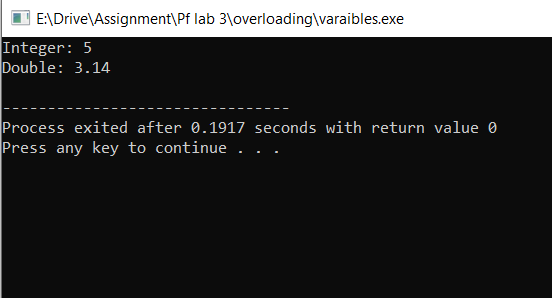
int main() {

print(5);

print(3.14);

return 0;

} OUTPUT:



Code 5:

#include<iostream>

using namespace std;

void print(int x)

{

cout<<"integer value :"<<x<<endl;

}

void print(float x)

{

cout<<"floating poin value :"<<x<<endl;

}

void print(string s1)

{

cout<<"string ="<<s1<<endl;

}

void print(char \*s1)

{

cout<<"character array :"<<s1<<endl;

}

int main()

{

print(5);

print(8.7f);

string s="hello word ";

print(s);

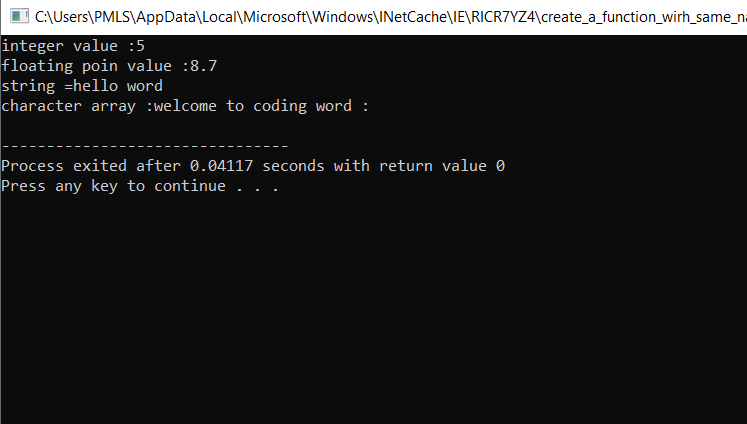
char str[40]="welcome to coding word :";

print(str);

return 0;

}

OUTPut:



**TEMPLET FUNNCTION**

CODE 1:

#include<iostream>

using namespace std;

template<class t>

t MAX(t a,t b)

{

if(a>b)

{

return a;

}

else

{

return b;

}

}

int main()

{

int a=7,b=9;

cout<<MAX(7,9)<<endl;

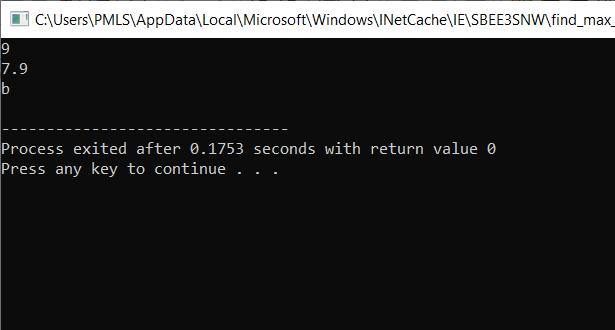
cout<<MAX(5.8f,7.9f)<<endl;

cout<<MAX('a','b')<<endl;

return 0;

}

OUTPUT:



Code 2:

#include<iostream>

using namespace std;

template <class t>

void printstring(t string)

{

cout<<string;

}

using namespace std;

int main()

{

string str1="i love pakistan!";

char str2[30]="c++ is powerful language :";

cout<<"string =";

printstring(str1);

cout<<endl;

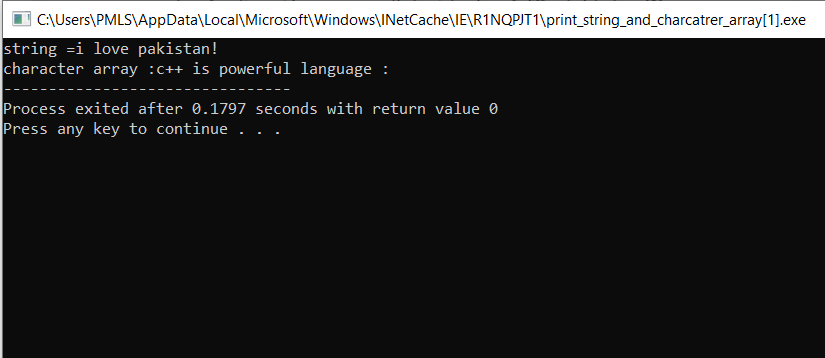
cout<<"character array :";

printstring(str2);

return 0;

}

OUTPUT:



Code 3:

#include<iostream>

using namespace std;

template <class t>

void printarray(t array[],int size)

{

for(int i=0;i<size;i++)

{

cout<<array[i]<<" ";

}

cout<<endl;

}

using namespace std;

int main()

{

double array[10]={1,2,3,4,5,6,7,8,9,10};

cout<<"double array :";

printarray(array,10);

int array2[10]={1,2,3,4,5,6,7,8,9,10};

cout<<"integer array :";

printarray(array2,10);

float array3[10]={1.4f,2.5f,3.7f,4.8f,5.9f,6.1f,7.3f,8.1f,9.2f,10.7f};

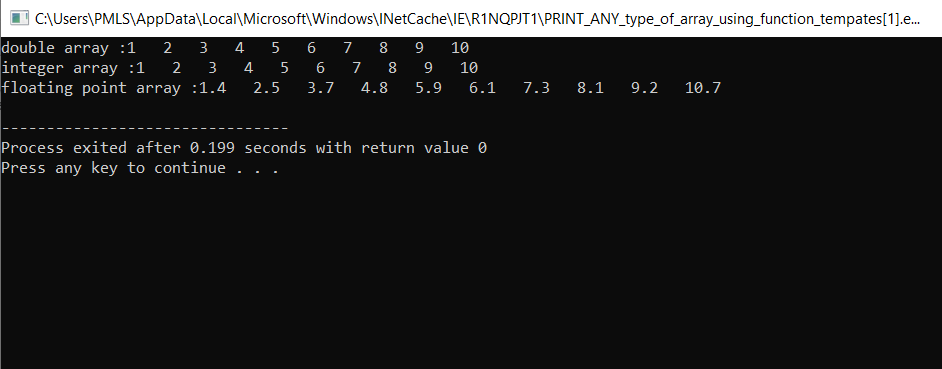
cout<<"floating point array :";

printarray(array3,10);

return 0;

}

OUTPUT:



Code 4:

#include<iostream>

using namespace std;

template <typename N>

N max(N x, N y, N z)

{

int max =x;

if(max<y){

max=y;

if(max<z)

{

max=z;

}

}

else if(max<z)

{

max=z;

if(max<y)

{

max=y;

}

}

return max;

}

int main()

{

int a=4, b=7, c=9;

cout<<"Maximum value is" <<max(a,b,c)<<endl;

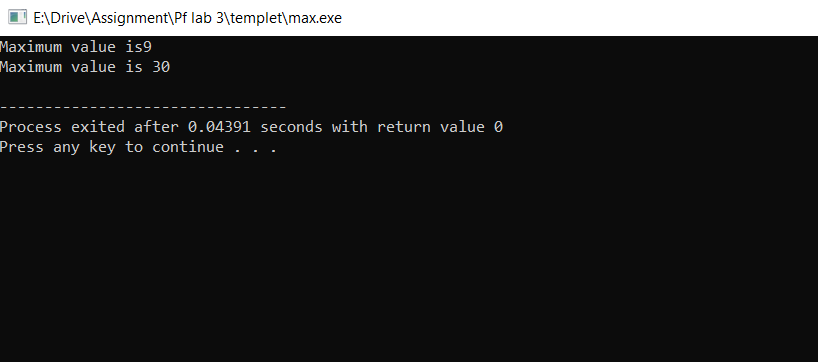
double x=20.5,y=30.7,z=9.5;

cout<<"Maximum value is "<<max(x,y,z)<<endl;

return 0;

}

OutPut:



Code 5:

#include<iostream>

using namespace std;

template <typename T>

T Sum(T x, T y, T z)

{

return x+y+z;

}

int main()

{

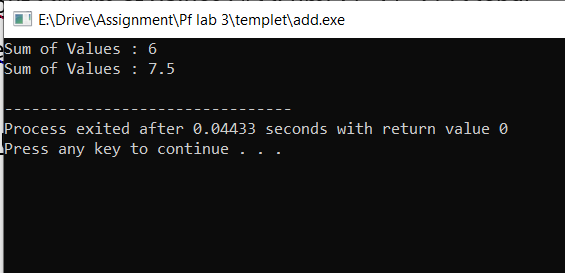
cout<<"Sum of Values : "<<Sum(2,2,2) <<endl;

cout<<"Sum of Values : "<<Sum(2.5,2.5,2.5)<<endl;

return 0;

}

OUTPUT:



**Pointers**

CODE 1:

#include<iostream>

using namespace std;

int main()

{

int x=10;

float y=6.7;

void \*p1,\*p2;

p1=&x;

p2=&y;

\*((int\*)p1)=20;

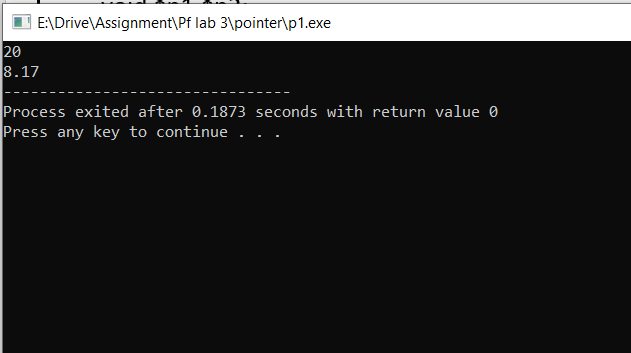
cout<<x<<endl;

\*((float\*)p2)=8.17;

cout<<y;

return 0;}

Output:



Code 2:

#include<iostream>

using namespace std;

void swap(int \*\*\*p,int \*q)

{

int temp;

temp=\*\*\*p;

\*\*\*p=\*q;

\*q=temp ;

}

int main()

{

int A=10, \*p1, \*\*p2, \*\*\*p3,B=20 ,\*n;

p1=&A;

p2=&p1;

p3=&p2;

n=&B;

swap(p3,n);

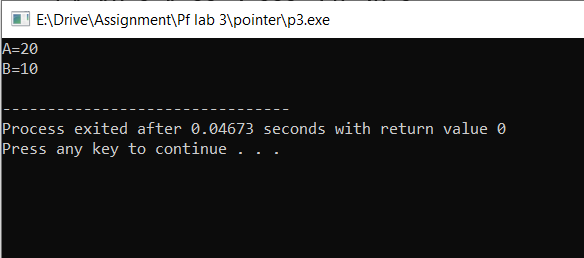
cout<<"A="<<A<<endl;

cout<<"B="<<B<<endl;

return 0;

}

Output:



Code 3:

#include<iostream>

using namespace std;

void sort(int \*s1);

int main()

{

int A[5]={6,1,9,2,3},i;

sort(A);

for(i=0;i<5;i++)

{

cout<<A[i]<<" ";

}

return 0;

}

void sort(int \*s1)

{

int \*s2,i,j,temp;

for(i=0;i<5;i++)

{

s2=s1;

for(j=i+1;j<5;j++)

{

s2++;

if(\*s1<\*s2)

{

temp=\*s1;

\*s1=\*s2;

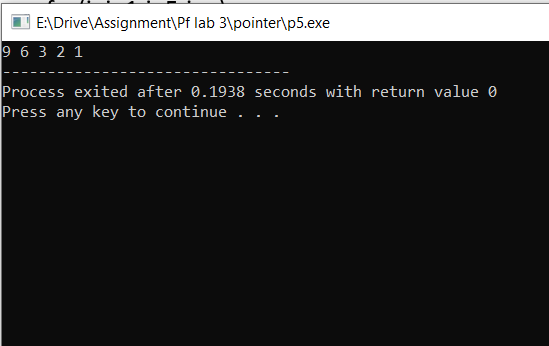
\*s2=temp;

}

}

s1++;

} Output:

}

Code 4:

#include<iostream>

using namespace std;

int\* Multiply(int \*s)

{

for (int i =0; i<5; i++)

{

\*(s+i)= \*(s+i) \* 5;

}

return s;

}

int main()

{

int arr[5] = {1, 2, 3, 4, 5};

int \*s;

s= Multiply(arr);

for (int i=0; i<5; i++)

{

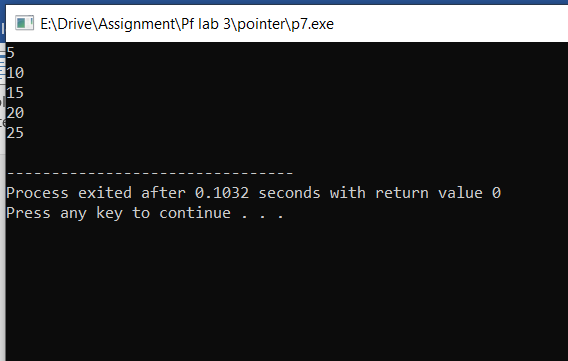
cout<<\*(s+i)<<endl;

}

return 0;

}

Output:



Code 5:

#include <iostream>

using namespace std;

void Ispalindrome(char \*s1)

{

int len=0, flag=1,i;

char \*s2;

s2 = s1;

for(i=0; \*s1!='\0'; i++)

{

len++;

s1++;

}

s1--;

for (i=0; i<len/2; i++)

{

if (\*s2 != \*s1)

{

flag=0;

break;

}

s1--;

s2++;

}

if (flag)

{

cout<<"It is palindrome \n";

}

}

int main ()

{

char str[20];

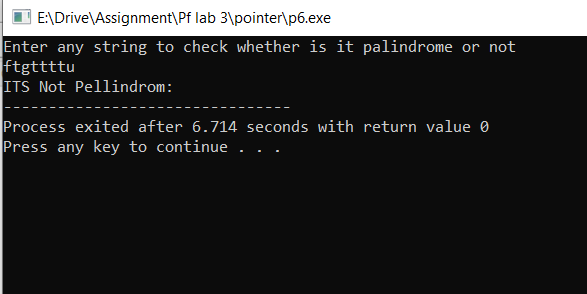
cout<<"Enter any string to check whether is it palindrome or not\n";

cin>>str;

Ispalindrome(str);

return 0;

} Output

: 

Code 6:

#include <iostream>

using namespace std;

int main()

{

char str[20] = "Computer";

char \*p;

p = str;

cout<<"String Berfore Reverse\n";

for (int i=0; \*p!='\0'; i++)

{

cout<<\*p;

p++;

}

cout<<endl;

cout<<"String After Reverse is \n";

p--;

for (int i=0; \*p!='\0'; i++)

{

cout<<\*p;

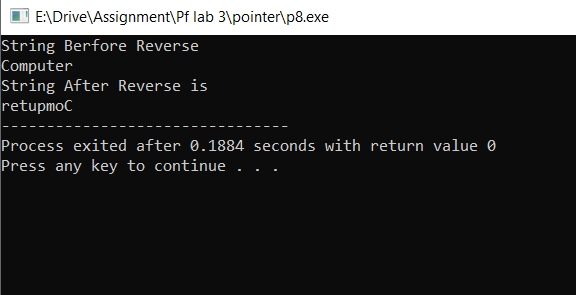
p--;

}

return 0;

}

OutPUT:



Code 7:

#include <iostream>

using namespace std;

void Copy(char \*s2, char \*s1)

{

for (int i=0; \*s1!='\0'; i++)

{

\*s2 = \*s1;

s2++;

s1++;

}

\*s2 = '\0';

}

int main()

{

char str1[20] = "Computer", str2[20];

cout<<"First String is "<<str1<<endl;

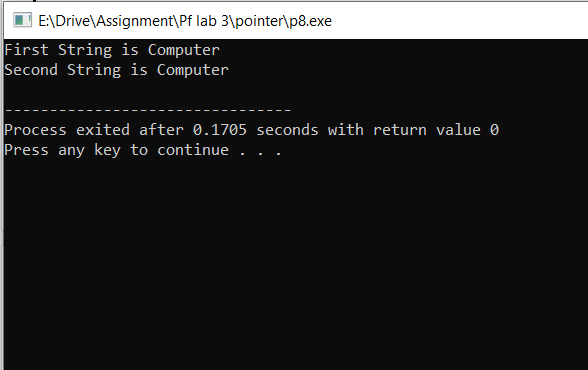
Copy(str2, str1);

cout<<"Second String is "<<str2<<endl;

return 0;

}

Output



Code 8:

#include <iostream>

using namespace std;

int main()

{

int \*p1, \*p2, sum;

p1 = new int;

p2 = new int;

cout<<"Enter First values \n";

cin>>\*p1;

cout<<"Enter Second values \n";

cin>>\*p2;

sum = \*p1 + \*p2;

cout<<"sum of two values = "<<sum;

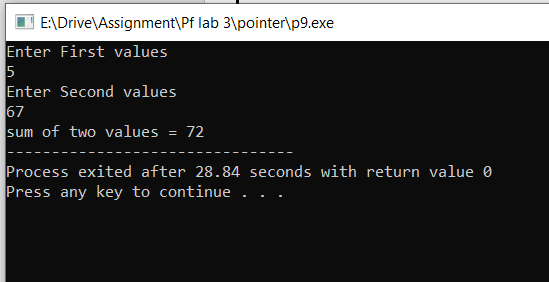
delete p1;

delete p2;

return 0;

}

Output:



Code 9:

#include<iostream>

using namespace std;

int main()

{

int x=10,y=20,\*p1,\*p2;

p1=&x;

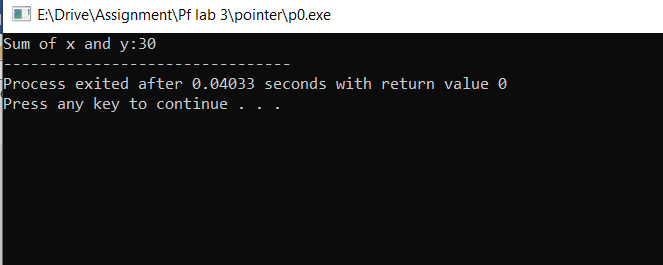
p2=&y;

cout<<"Sum of x and y:"<<\*p1+\*p2;

return 0;

}

Output:



Code 10:

#include<iostream>

using namespace std;

int main()

{

int x=10,y,\*p;

p=&y;

cout<<"Enter value:";

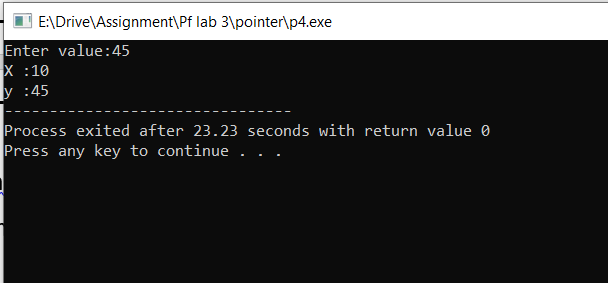
cin>>\*p;

cout<<"X :"<<x<<endl;

cout<<"y :"<<y;

return 0;}

Output:



**FUNCTION ARRAY**

CODE 1:

#include<iostream>

using namespace std;

int maxelement(int array[],int size)

{

int max=array[0];

for(int i=1;i<size;i++)

{

if(array[i]>max)

{

max=array[i];

}

}

return max;

}

int main()

{

int size=10;

int array[size]={1,2,3,4,5,6,7,8,9,10};

cout<<"array =:";

for(int i=0;i<size;i++)

{

cout<<array[i]<<" ";

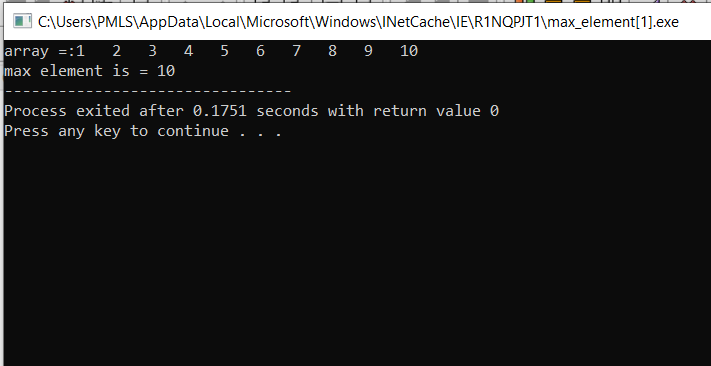
}

cout<<endl;

cout<<"max element is = "<<maxelement(array,size);

return 0;

}

OUTPUT: 

CODE 2:

#include<iostream>

using namespace std;

void searchelement(int array[],int sv,int size)

{

int flag=0,i=0;

for( i=0;i<size;i++)

{

if(array[i]==sv)

{

flag=1;

break;

}

}

if(flag)

{

int ap=0;

for( int j=0;j<size;j++)

{

if(array[j]==sv)

{

ap=ap+1;

}

}

cout<<sv<<" value have first apperance on index : "<<i<<endl;

cout<<"and it is appear "<<ap-1<<" more time in array :";

}

else

{

cout<<"no value found in array : ";

}

}

int main()

{

int array[10]={1,2,3,4,5,6,8,8,2,8},sn;

cout<<"ARRay:";

for(int i=0;i<10;i++)

{

cout<<array[i]<<" ";

}

cout<<endl;

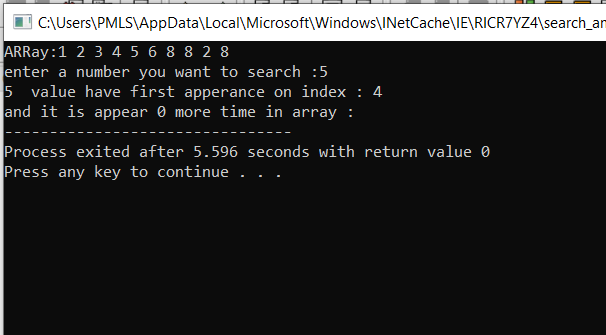
cout<<"enter a number you want to search :";

cin>>sn;

searchelement(array,sn,10);

return 0;

} OUTPUT:

: 

Code 3:

#include <iostream>

using namespace std;

void printArray(int arr[], int size) {

for (int i = 0; i < size; ++i) {

cout << arr[i] << " ";

}

cout << endl;

}

int main() {

int myArray[] = {1, 2, 3, 4, 5};

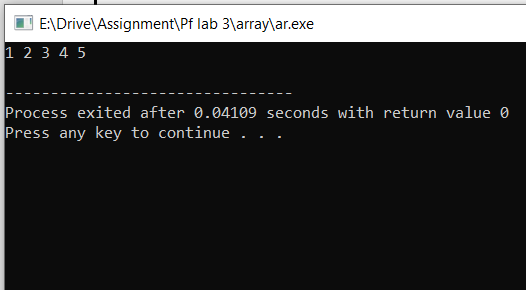
int size = sizeof(myArray) / sizeof(myArray[0]);

printArray(myArray, size);

return 0;

}

OUTPUt:



Code 4:

#include <iostream>

using namespace std;

void printArray(int arr[][3], int rows, int cols) {

for (int i = 0; i < rows; ++i) {

for (int j = 0; j < cols; ++j) {

cout << arr[i][j] << " ";

}

cout << endl;

}

}

int main() {

int myArray[][3] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

int rows = sizeof(myArray) / sizeof(myArray[0]);

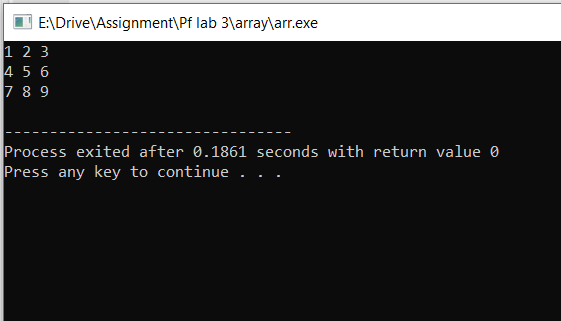
int cols = sizeof(myArray[0]) / sizeof(myArray[0][0]);

printArray(myArray, rows, cols);

return 0;

}

Output:



CODE 5:

#include <iostream>

using namespace std;

// declare function to display marks

// take a 1d array as parameter

void display(int m[5]) {

cout << "Displaying marks: " << endl;

// display array elements

for (int i = 0; i < 5; ++i) {

cout << "Student " << i + 1 << ": " << m[i] << endl;

}

}

int main() {

// declare and initialize an array

int marks[5] = {88, 76, 90, 61, 69};

// call display function

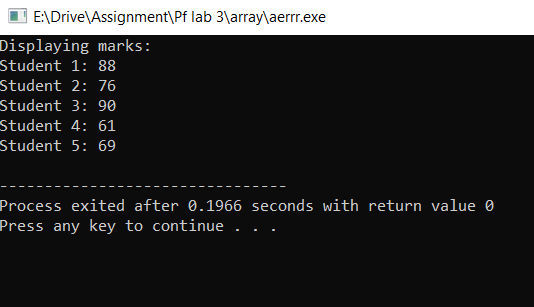
// pass array as argument

display(marks);

return 0;

}

OUTPUT:



Code 6:

// C++ Program to display the elements of two

// dimensional array by passing it to a function

#include <iostream>

using namespace std;

// define a function

// pass a 2d array as a parameter

void display(int n[][2]) {

cout << "Displaying Values: " << endl;

for (int i = 0; i < 3; ++i) {

for (int j = 0; j < 2; ++j) {

cout << "num[" << i << "][" << j << "]: " << n[i][j] << endl;

}

}

}

int main() {

// initialize 2d array

int num[3][2] = {

{3, 4},

{9, 5},

{7, 1}

};

// call the function

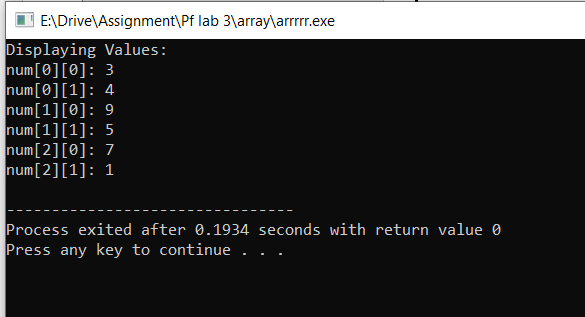
// pass a 2d array as an argument

display(num);

return 0;

}

Output:



Code 7:

#include<iostream>

using namespace std;

int Sum(int x[])

{

int s=0;

for (int i=0; i<5;i++)

s=s+x[i];

return s;

}

int main()

{

int arr[5] = {2, 3, 4, 5, 7}, i;

for (int i=0;i<5;i++)

cout<<arr[i]<<"\t";

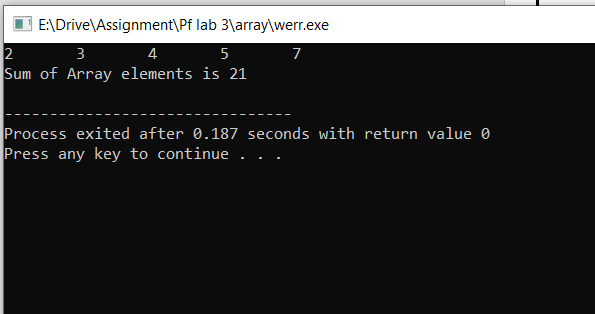
cout<<"\nSum of Array elements is "<<Sum(arr);

cout<<endl;

return 0;

}

Output:



Code 8:

#include<iostream>

#include<conio.h>

using namespace std;

int main()

{

int arr[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

int i, sum=0;

for (i=0; i<=9; i++)

{

if (arr[i]%2==0)

{

cout<<arr[i]<<endl;

sum = sum+arr[i];

}

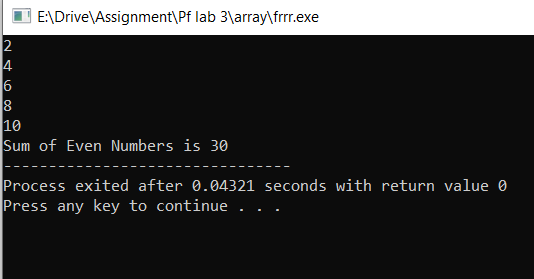
}

cout<<"Sum of Even Numbers is "<<sum;

return 0;

}

Output:



Code 9:

//1:Program to find list of Even numbers and their sum by passing

//2-D to a function.

#include <iostream>

using namespace std;

int EvenSum(int arrnum[3][3], int ArrEvn[9]);

int main()

{

int arr[3][3] = {16, 21, 323, 44, 59, 56,37, 78,49}, even[9];

int i, j, size, sum=0;

size = EvenSum(arr, even);

cout<<"passed values of 2-D to function are \n";

for (i=0; i<3; i++)

{

for (j=0; j<3; j++)

{

cout<<arr[i][j]<<"\t";

}

cout<<endl;

}

cout<<"Even values of 2-D and their sum are \n";

for (i=0; i<size; i++)

{

cout<<even[i]<<"\t";

sum = sum+even[i];

}

cout<<"Sume of Even values is "<<sum<<endl;

return 0;

}

int EvenSum(int arrnum[3][3], int ArrEvn[9])

{

int s=0, i, j;

for (i=0; i<3; i++)

{

for (j=0; j<3; j++)

{

if (arrnum[i][j]%2==0)

{

ArrEvn[s]=arrnum[i][j];

s++;

}

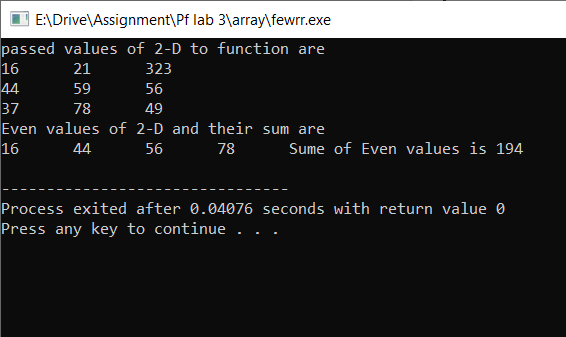
}

}

return s;

}

Output:



Code 10:

#include<iostream>

using namespace std;

int odd(int arr[3][2]);

int main()

{

int i,j,od;

int number[3][2];

for(i=0;i<3;i++)

for(j=0;j<2;j++)

{

cout<<"Enter the values of number["<<i<<"]["<<j<<"]";

cin>>number[i][j];

}

od=odd(number);

cout<<"odf number value is:"<<od<<endl;

}

int odd(int arr[3][2])

{

int i,j;

for(i=0;i<3;i++)

for(j=0;j<2;j++)

{

if(arr[i][j]%2!=0)

return arr[i][j];

}

}

Output:

