**Assignment-24 Solution NAME : OM PANT**

**Functions in C++**

1. Define a function to check whether a given number is a Prime number or not.

Ans-

// 1. Define a function to check whether a given number is a Prime number or not.

#include<iostream>

using namespace std;

int checkPrime(int num){

    int i,flag=1;

    for(i=2;i<=num/2;i++){

            if(num%i == 0){

                flag = 0;

                return 0;

            }

    }

    if(flag==1){

        return 1;

    }

}

int main(){

    int num,i,result;

    cout<<"Enter a number"<<endl;

    cin>>num;

    result = checkPrime(num);

    if(result){

        cout<<"Prime Number"<<endl;

    }

    else{

        cout<<"Not a Prime Number"<<endl;

    }

    return 0;

}

1. Define a function to find the highest value digit in a given number.

Ans-

// 2. Define a function to find the highest value digit in a given number.

#include<iostream>

using namespace std;

int highestDigit(int number){

    int max =0;

    if(number<0)

        number \*= (-1);

    while(number%10 != 0){

        if(number%10 > max)

            max = number%10;

        number = number/10;

    }

    return max;

}

int main(){

    int number;

    cout<<"Enter a number"<<endl;

    cin>>number;

    cout<<"Highest Digit in Number: "<<highestDigit(number)<<endl;

    return 0;

}

1. Define a function to calculate x raised to the power y.

Ans-

// 3. Define a function to calculate x raised to the power y.

#include<iostream>

#include<cmath>

using namespace std;

void power(int x,int y){

    cout<<x<<" raised to the power "<<y<<" is: "<<pow(x,y);

}

int main(){

    int x,y;

    cout<<"Enter Base value: ";

    cin>>x;

    cout<<"Enter Exponent Value: ";

    cin>>y;

    power(x,y);

    return 0;

}

1. Define a function to print Pascal Triangle up to N lines.

Ans-

// 4. Define a function to print Pascal Triangle up to N lines.

#include<iostream>

using namespace std;

void PascalTriangle(int rows){

 for (int i = 0; i < rows; i++)

    {

        int val = 1;

        for (int j = 1; j < (rows - i); j++)

        {

            cout << "   ";

        }

        for (int k = 0; k <= i; k++)

        {

            cout << "     " << val;

            val = val \* (i - k) / (k + 1);

        }

        cout << endl << endl;

    }

}

int main()

{

    int rows;

    cout << "Enter the number of rows : ";

    cin >> rows;

    cout << endl;

    PascalTriangle(rows);

    cout << endl;

    return 0;

}

1. Define a function to check whether a given number is a term in a Fibonacci series or not.

Ans-

// 5. Define a function to check whether a given number is a term in a Fibonacci series or not.

#include <bits/stdc++.h>

using namespace std;

bool isPerfectSquare(int x)

{

    int s = sqrt(x);

    return (s \* s == x);

}

bool isFibonacci(int n)

{

    return isPerfectSquare(5 \* n \* n + 4)

        || isPerfectSquare(5 \* n \* n - 4);

}

int main()

{

    int num;

    cout<<"Enter a number: ";

    cin>>num;

    isFibonacci(num)? cout << num << " is a Fibonacci Number \n"

            : cout << num << " is a not Fibonacci Number \n";

    return 0;

}

1. Define a function to swap data of two int variables using call by reference

Ans-

// 6. Define a function to swap data of two int variables using call by reference.

#include<iostream>

using namespace std;

void swap(int &a, int &b){  //Reference variables

    int temp;

    temp = a;

    a = b;

    b = temp;

}

int main(){

    int x,y;

    cout<<"Enter two values: ";

    cin>>x>>y;

    cout<<"Before Swap X: " <<x<<" Y: "<< y << endl;

    swap(x,y);

    cout<<"After Swap X: " <<x<<" Y: "<< y << endl;

    return 0;

}

1. Write a function using the default argument that is able to add 2 or 3 numbers.

Ans-

// 7. Write a function using the default argument that is able to add 2 or 3 numbers.

#include<iostream>

using namespace std;

int add(int x,int y,int z=0){

    return x+y+z;

}

int main(){

    cout<<"Addition: (Two Numbers) 2,3: "<<add(2,3)<<endl;

    cout<<"Addition: (Three Numbers) 2,5,7: "<<add(2,5,7)<<endl;

    return 0;

}

1. Define overloaded functions to calculate area of circle, area of rectangle and area of triangle

Ans-

// 8. Define overloaded functions to calculate area of circle, area of rectangle and area of  triangle

#include<iostream>

using namespace std;

int area(float rad){

    return 3.14\*rad\*rad;

}

int area(float len,float wid){

    return len\*wid;

}

int area(int base,int heigth){

    return (base\*heigth)/2;

}

int main(){

    int rad,len,wid,base,heigth,choice;

    cout<<"Enter your Choice:\n 1: For area of Circle\n 2: For Area of rectangle\n 3: Area of triangle"<<endl;

    cin>>choice;

    switch (choice)

    {

    case 1:

        cout<<"Enter radius of circle: ";

        cin>>rad;

        cout<<"Area is: "<<area(rad)<<endl;;

        break;

    case 2:

        cout<<"Enter Length of Rectangle: ";

        cin>>len;

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

        cin>>wid;

        cout<<"Area is: "<<area(len,wid)<<endl;

        break;

    case 3:

        cout<<"Enter base of Triangle: ";

        cin>>base;

        cout<<endl<<"Enter height of Triangle: ";

        cin>>heigth;

        cout<<"Area is: "<<area(base,heigth)<<endl;

        break;

    default:

        cout<<"Enter a valid choice!!"<<endl;

        break;

    }

    return 0;

}

1. Write functions using function overloading to find a maximum of two numbers and both the numbers can be integer or real.

Ans-

// 9. Write functions using function overloading to find a maximum of two numbers and both  the numbers can be integer or real.

#include<iostream>

using namespace std;

int max(int n1,int n2){

    return n1>n2?n1:n2;

}

float max(int n1,float n2){

    return n1>n2?n1:n2;

}

float max(float n1,int n2){

    return n1>n2?n1:n2;

}

float max(float n1,float n2){

    return n1>n2?n1:n2;

}

int main(){

    float n1,n2;

    cout<<"Enter two numbers"<<endl;

    cin>>n1>>n2;

    cout<<"Maximum of two is: "<<max(n1,n2)<<endl;

    return 0;

}

1. Write functions using function overloading to add two numbers having different data types.

Ans-

// 10. Write functions using function overloading to add two numbers having different data types.

#include<iostream>

using namespace std;

int add(int n1,int n2){

    return n1+n2;

}

float add(float n1,int n2){

    return n1+n2;

}

float add(int n1,float n2){

    return n1+n2;

}

float add(float n1,float n2){

    return n1+n2;

}

int main(){

    float n1,n2;

    cout<<"Enter two numbers"<<endl;

    cin>>n1>>n2;

    cout<<"Addition of two is: "<<add(n1,n2)<<endl;

    return 0;

}