**Assignment 1**

**Fundamentals of Programming**

**CS-114**

**FALL 2023**

1. Write a C++ program to display factors of a number using for loops.

#include<iostream>

using namespace std;

int main(){

int number;

cout<<"enter a number: ";

cin>>number;

cout<<"factors of "<<number<<": ";

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

if(number%i==0){

cout<<i<<" ";

}

}

}



1. Write output to the following code.

|  |
| --- |
| #include <iostream>    int main() { int x = 5;  int y = 10;    if (x == 5) if (y == 10)  std::cout << "x is 5 and y is 10" << std::endl; else  std::cout << "x is not 5" << std::endl;    return 0;  } |

OUTPUT:

x is 5 and y is 10

1. Write a C++ program, take an integer value from user and check if it’s greater than 10 and less than equal to 20. Print 1 if yes and print 0 if no. Use appropriate datatype for output.

#include <iostream>

using namespace std;

int main(){

int userInput;

cout<<"Enter an integer: ";

cin>>userInput;

if(userInput>10&&userInput<=20){

cout<<"1"<<endl;

}

else{

cout<<"0"<<endl;

}

}



1. Write a C++ program that uses a **while** loop to find the largest prime number less than a given positive integer **N**. Your program should take the value of **N** as input from the user and then find the largest prime number less than or equal to **N**. You are not allowed to use any library or pre-existing functions to check for prime numbers.

#include <iostream>

using namespace std;

int main(){

int N;

cout<<"enter a positive integer N: ";

cin>>N;

while(N>1){

bool isPrime=true;

for(int i=2;i\*i<=N;++i){

if (N%i==0){

isPrime=false;

break;

}

}

if(isPrime){

cout<<"largest prime number less than or equal to N: "<<N<<endl;

break;

}

--N;

}

}



1. Write a C++ program, take two string as input from user and check if both strings are equal or not. If they are equal make them unequal by rotating string. e.g., Hello is turned into olleH etc.

#include <iostream>

#include <string>

using namespace std;

int main(){

string str1,str2;

cout<<"enter the first string: ";

cin>>str1;

cout<<"enter the second string: ";

cin>>str2;

if(str1==str2){

for (int i=0;i<str1.length()/2;++i){

char temp=str1[i];

str1[i]=str1[str1.length()-i-1];

str1[str1.length()-i-1]=temp;

}

cout<<"inverted string 1: "<<str1<<endl;

cout<<"original string 2: "<<str2<<endl;

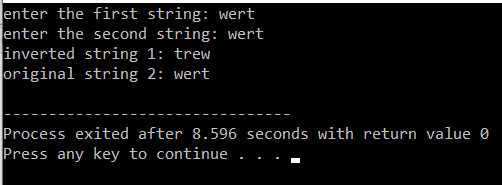
}

else{

cout<<"strings are not equal"<<endl;

}

}



1. Perform division in C++ **without /** using **for** loops. You can use **/** only to display the final results. Your dividend must be greater than divisor.

#include <iostream>

using namespace std;

int main(){

int dividend,divisor,remainder;

cout<<"enter dividend: ";

cin>>dividend;

cout<<"enter divisor: ";

cin>>divisor;

int quotient=0;

if(dividend<divisor){

cout<<"dividend must be greater than divisor."<<endl;

}

else{

while(dividend>=divisor){

dividend-=divisor;

quotient++;

}

cout<<"quotient: "<<quotient<<endl;

cout<<"remainder: "<<dividend<<endl;

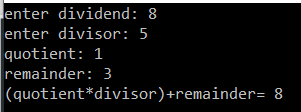
remainder=dividend;

}

dividend=(quotient\*divisor)+remainder;

cout<<"(quotient\*divisor)+remainder= "<<dividend;

}



1. Write a C++program for a string which may contain lowercase and uppercase characters. The task is to remove all duplicate characters from the string and find the resultant string.

#include <iostream>

#include <string>

using namespace std;

int main(){

string input;

cout<<"Enter a string: ";

cin>>input;

string result="";

for(int i=0;i<input.length();++i){

char currentChar=input[i];

bool isDuplicate=false;

for(int j=0;j<result.length();++j){

if(result[j]==currentChar){

isDuplicate=true;

break;

}

}

if(!isDuplicate){

result+=currentChar;

}

}

cout<<"Resultant string after removing duplicates: "<<result<<endl;

}



1. Suppose an integer array a[5] = {1,2,3,4,5}. Add more elements to it and display them in C++.

#include <iostream>

using namespace std;

int main(){

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

int n=5-1,num;

cout<<"do you want to add a number to the array?(1=yes, 0=no)"<<endl;

bool add;

cin>>add;

while(add){

cout<<"enter number to add to array: "<<endl;

cin>>num;

a[n+1]=num;

cout<<"do you want to continue adding numbers to the array?(1=yes, 0=no)"<<endl;

cin>>add;

}

cout<<"Array elements: ";

for(int i=0;i<8;++i){

cout<<a[i]<<" ";

}

cout<<endl;

}

1. Given an integer array and an integer **X**. Find if there’s a triplet in the array which sums up to the given integer **X**.

#include <iostream>

using namespace std;

int main(){

int n;

int arr[10];

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

cout<<"enter integer for serial number "<<(i+1)<<": ";

cin>>arr[i];

}

n=10;

int X;

cout<<"enter the number to find sum of: ";

cin>>X;

bool tripletFound=false;

cout<<"triplets with sum "<<X<<" and no repetition:"<<endl;

for(int i=0;i<n-2;++i){

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

for(int k=j+1;k<n;++k){

if(arr[i]+arr[j]+arr[k]==X&&arr[i]!=arr[j]&&arr[j]!=arr[k]&&arr[i]!=arr[k]){

cout<<arr[i]<<", "<<arr[j]<<", "<<arr[k]<<endl;

tripletFound=true;

}

}

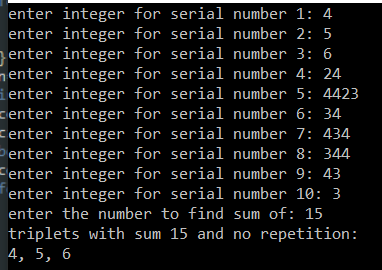
}

}

if(!tripletFound){

cout<<"no triplet found with the sum "<<X<<" and no repetition."<<endl;

}

}

1. Implement Bubble Sort on an array of 6 integers.

#include <iostream>

using namespace std;

int main(){

int arr[6];

for(int i=0;i<6;++i){

cout<<"enter integer for serial number "<<(i+1)<<": ";

cin>>arr[i];

}

cout<<"Original Array is: ";

for (int i=0;i<6;++i){

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

}

cout<<endl;

for(int i=0;i<6-1;++i){

for(int j=0;j<6-i-1;++j){

if(arr[j]>arr[j+1]){

int temp=arr[j];

arr[j]=arr[j+1];

arr[j+1]=temp;

}

}

}

cout<<"Sorted array: ";

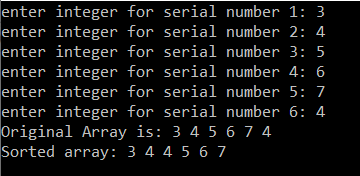
for(int i=0;i<6;++i){

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

}

cout<<endl;

}

****

**Rename file as yourname\_RollNumber\_Assignment-1\_Section\_X.pdf.**