

A blue text on a white background

Description automatically generated

**DEPARTMENT OF MECHANICAL ENGINEERING  
  
Subject: Fundamentals of Programming  
Assignment No. 1  
Submitted by: Zahoor Azam  
Registration number: 453972  
Semester No. 1  
Date: November 20, 2023**

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

**Solution**#include <iostream>

using namespace std;

int main()

{

int num;

cout << "Enter the number: ";

cin>>num;

cout<<"Factors of "<<num<<endl;

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

if (num%i==0) {

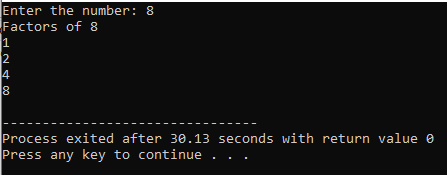
cout<<i<<endl;

}

}

return 0;

}



1. **Write output to the following code.**

**#include <iostream>**

**Using namespace std;**

**int main() {**

**int x = 5;**

**int y = 10;**

**if (x == 5)**

**if (y == 10)**

**cout << "x is 5 and y is 10" << endl;**

**else**

**cout << "x is not 5" << endl;**

**return 0;**

**}**

**Solution**

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.**

**Solution**

#include <iostream>

using namespace std;

int main()

{

int num;

cout<<"Enter the number: ";

cin>>num;

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

cout<<"1";

}

else {

cout<<"0";

}

return 0;

}

A black screen with white text

Description automatically generated

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.**

**Solution**

#include <iostream>

using namespace std;

int main()

{

int N, factors, i, j;

cout<<"Enter value of N: ";

cin>>N;

i = N;

while (i >= 2) {

factors = 0;

j = 1;

while (j <= i) {

if (i%j==0) {

factors++;

}

j++;

}

if (factors==2) {

cout<<"The highest prime number lower than N is: "<<i;

break;

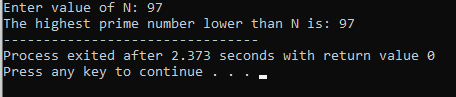
}

i--;

}

return 0;

}



**5. 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**

**Solution**

#include <iostream>

#include <string>

using namespace std;

int main()

{

string str1, str2, reversed;

reversed = "";

cout<<"Enter 1st String: ";

cin>>str1;

cout<<"Enter 2nd String: ";

cin>>str2;

if (str1 == str2) {

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

reversed = str1[i] + reversed;

}

cout<<"Strings are equal. Reversed string is: ";

cout<<reversed;

}

else {

cout<<"Strings are unequal";

}

return 0;

}

A black screen with white text

Description automatically generated

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.**

**Solution**

#include <iostream>

using namespace std;

int main()

{

int dividend, divisor, remainder, quotient;

cout<<"Enter dividend: ";

cin>>dividend;

cout<<"Enter divisor: ";

cin>>divisor;

if (dividend < divisor) {

cout<<"Please Enter a dividend greater than divisor.";

return 1;

}

remainder = dividend;

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

remainder -= divisor;

if (remainder < divisor) {

quotient = i;

break;

}

}

cout<<dividend<<" / "<<divisor<<" = "<<quotient;

return 0;

}

A black screen with white text

Description automatically generated

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.**

**Solution**

#include <iostream>

#include <string>

using namespace std;

int main()

{

string str, result;

bool found;

cout<<"Enter String: ";

cin>>str;

result = "";

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

found = false;

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

if ( str[i] == result[j] ) {

found = true;

}

}

if (found == false) {

result += str[i];

}

}

cout<<"Resultant String: "<<result;

return 0;

}

A black screen with white text

Description automatically generated

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

**Solution**

#include <iostream>

using namespace std;

int main()

{

int newA[8], a[5] = {1,2,3,4,5};

cout<<"Current array is: {";

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

cout<<a[i];

if (i==4)

continue;

cout<<", ";

}

cout<<"}\n";

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

newA[i] = a[i];

}

cout<<"Enter 3 integers which will be added to the array: "<<endl;

for (int i = 5; i<8; i++)

cin>>newA[i];

cout<<"Printing elements of array"<<endl;

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

cout<<newA[i]<<" ";

}

return 0;

}

A screenshot of a computer program

Description automatically generated

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.**

**Solution**

#include <iostream>

using namespace std;

int main()

{

int a, b, c, X, sum, arr[10];

cout<<"Enter 10 integers for array: ";

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

cin>>arr[i];

}

bool found = false;

cout<<"Enter integer X: ";

cin>>X;

cout<<"Triplets are: ";

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

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

if (i == j)

continue;

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

if (z == i || z == j)

continue;

sum = arr[i] + arr[j] + arr[z];

if (sum == X) {

cout<<" ("<<arr[i]<<", "<<arr[j]<<", "<<arr[z]<<")";

found = true;

}

}

}

}

if (found == false) {

cout<<"No Triplet Found";

}

return 0;

}

A screenshot of a computer

Description automatically generated

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

**Solution**

#include <iostream>

using namespace std;

int main()

{

int a, s = 6, arr[s];

cout<<"Enter "<<s<<" integers for array: ";

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

cin>>arr[i];

}

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

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

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

a = arr[i];

arr[i] = arr[i+1];

arr[i+1] = a;

}

}

}

cout<<"Final Array: {";

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

cout<<arr[i];

if (i == s-1)

continue;

cout<<",";

}

cout<<"}";

return 0;

}

