**FOP ASSIGNMEMENT 1**

**name = muhammad shahzeb khan**

**registration number = 479677**

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

#include<iostream>

using namespace std;

int main (){

int number;// declared variables

cout<<"enter number whose factors are to be calculated "<<endl;

cin>>number;

int factors=1;// initialized a variable to value

cout<<"required factors of "<<" "<<number<<" "<<" are "<<endl;

while(factors<=number)// using while loop to get numbers after increment one by one upto input

{

if(number%factors==0)// using if condition to check which numbers are factors

{

cout<<factors<<endl;// printing output

}

factors++;// added increment of 1

}

}





**2. 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;**

**}**



x is 5 and y is 10

**3. 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 num;

cout<<"enter a number to be checked"<<endl;

cin>>num;

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

cout<<"1"<<endl;

}

else

{

cout<<"0"<<endl;

}

}





**4. 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 num, factors, i, j;

cout<<"Enter value of nuumber "<<endl;

cin>>num;

i = num;

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 number is "<<i<<endl;

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.**

#include <iostream>

#include <string>

using namespace std;

int main()

{

string x,y, reversed;

reversed = "";

cout<<"Enter 1st String"<<endl;

cin>>x;

cout<<"Enter 2nd String"<<endl;

cin>>y;

if (x == y) {

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

reversed = x[i] + reversed;

}

cout<<"both Strings are equal so the Reversed string is"<<endl;

cout<<reversed;

}

else {

cout<<"both Strings are not equal"<<endl;

}

return 0;

}





**6. 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, quotient;

cout<<"enter dividend which will be greater than divisor"<<endl;

cin>>dividend;

cout<<"Enter divisor"<<endl;

cin>>divisor;

if (dividend < divisor) {

cout<<"again enter a dividend greater than divisor"<<endl;

return 1;

}

remainder=dividend;

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

remainder=remainder-divisor;

if (remainder < divisor)

{

quotient = i;

break;

}

}

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

return 0;

}





**7. 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 a,result;

bool is;

cout<<"Enter String which is to be corrected"<<endl;

cin>>a;

result = "";

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

is= false;

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

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

is= true;

}

}

if (is== false) {

result += a[i];

}

}

cout<<"so the resultant corrected string will be "<<result<<endl;

return 0;

}





**8. 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 array2[8], array1[5] = {1,2,3,4,5};

cout<<"original array is {";

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

cout<<array1[i];

if (i==4)

continue;

cout<<", ";

}

cout<<"}"<<endl;

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

array2[i] = array1[i];

}

cout<<"enter the three additional numbers which are to be added"<<endl;

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

cin>>array2[i];

cout<<"the elements printed will be "<<endl;

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

cout<<array2[i]<<'\t';

}

return 0;

}





**9. 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 a,b,c,num,sum,array[10];

int i;

bool is;

cout<<"enter ten integers for making an array "<<endl;

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

cin>>array[i];

}

is = false;

cout<<"enter integer which must be the sum of components of triplets"<<endl;

cin>>num;

cout<<"so triplets will be "<<endl;

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 = array[i] + array[j] + array[z];

if (sum == num) {

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

is = true;

}

}

}

}

if (is == false) {

cout<<"there are no triplets"<<endl;

}

return 0;

}





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

#include<iostream>

using namespace std;

void bubbleSort(int arr[], int n) {

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

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

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

int temp = arr[j];

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

arr[j+1] = temp;

}

}

}

}

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

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

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

cout << endl;

}

int main() {

int arr[] = {64, 34, 25, 12, 22, 11, 90};

int n = sizeof(arr)/sizeof(arr[0]);

bubbleSort(arr, n);

cout<<"So the correct sorted array will be"<<endl;

printArray(arr, n);

return 0;

}



