**MUHAMMAD-RAYYAN**

**ME-15 SEC: B**

**FOP-LAB-ASSIGNMENT (1)**

**456847**

**SIR AFFAN**

#include<iostream>

#include<string>

using namespace std;

/\* Q.01

Factors of a number using FOR loop.\*/

/\*int main(){

int num,factor;

cout<<" Enter a natural number :";

cin>>num;

for(factor=1;factor<=num;factor++){

if(num%factor==0){

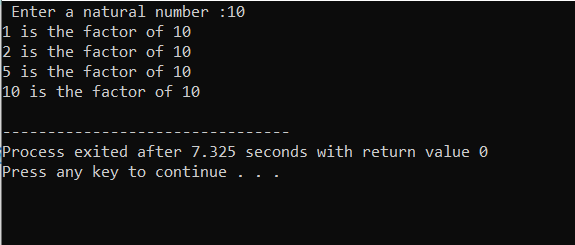
cout<<factor<<" is the factor of "<<num<<endl;

}

}

return 0;

}\*/



/\* Q.02

Showing Output.\*/

/\* int main(){

int x = 5;

int y = 10;

if (x == 5)

if (y == 10)

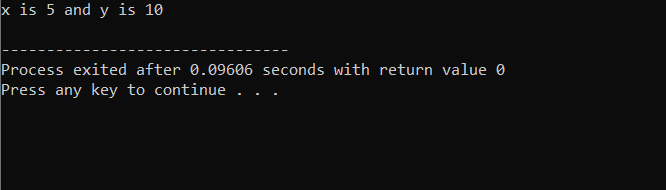
std::cout << "x is 5 and y is 10" << std::endl;

else

std::cout << "x is not 5" << std::endl;

return 0;

}\*/



/\* Q.03

Checking the integer either its greater than 10 & less and equal to 20 \*/

/\*int main(){

int number;

cout<<" Enter an integer : ";

cin>>number;

bool result = ( number > 10 && number <= 20);

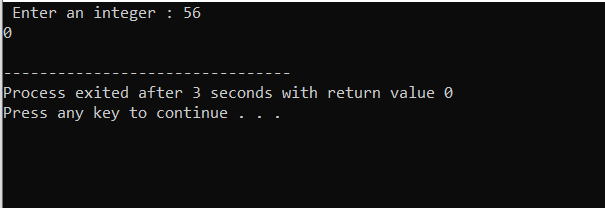
cout<<result<<endl;

// if output is 1 then it means given integer fulfills required condition.

// if output is 0 then it means given integer doesn't fulfills required condition.

return 0;

}\*/



/\* Q.04

Finding largest prime number less than or equal to given inputed integer N \*/

/\*int main(){

int num;

int divisor;

int a,b;

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

cin>>num;

a=num;

while(a>=2)

{ divisor=0;

b = 1;

while(a>=b)

{

if(a%b==0){

++divisor;

}

++b;

}

if(divisor==2){

cout<<" The highest prime number before the inputed number is : "<<a;

break;

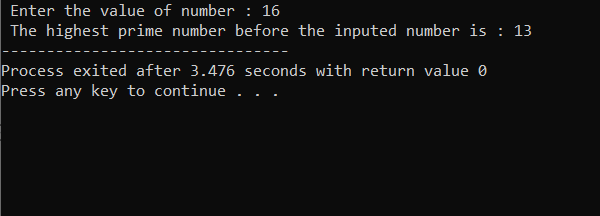
}

a--;

}

return 0;

}\*/



/\* Q.05

Checking Strings\*/

/\*int main(){

int i;

string x1 ;

string x2;

string rotated;

rotated = "";

cout<<"Enter the 1st String: ";

cin>>x1;

cout<<"Enter the 2nd String: ";

cin>>x2;

if (x1 == x2) {

for (i = 0; i < x1.length(); i++) {

rotated = x1[i] + rotated;

}

cout<<"Strings are equal so the rotated string is: ";

cout<<rotated;

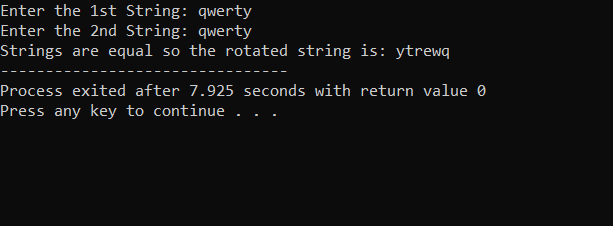
}

else {

cout<<"Strings are not equal";

}

return 0;



/\* Q.no 6

Performing division whitout using for loop.\*/

/\* int main(){

int divisor = 6;

int dividend = 36;

int quotient = 0;

while(dividend>=divisor)

{ dividend = dividend-divisor;

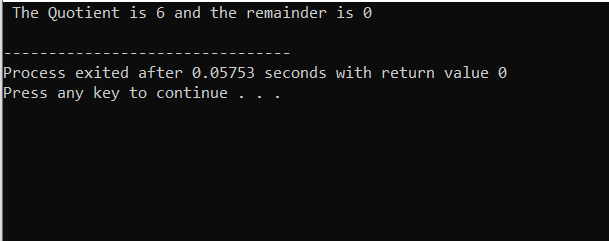
quotient++;

}

cout<<" The Quotient is "<<quotient<<" and the remainder is "<<dividend<<endl;

return 0;

} \*/



/\* Q.07

Removing duplicate characters\*/

/\* int main(){

string str;

string res="";

cout<<" Enter a string :"<<endl;

cin>>str;

int i;

int j;

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

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

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

break;

}

}

if(i==j){

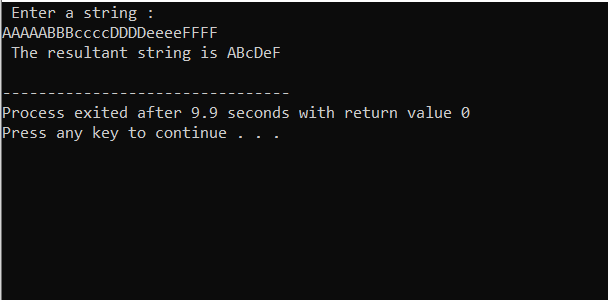
res+=str[i];

}

}

cout<<" The resultant string is "<<res<<endl;

}\*/



/\* Q.no 8

Adding elements in an array \*/

/\* int main (){

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

int newSize =8 ;

int news[] = {6,7,8};

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

{ orgs[i] = news[i-5]; }

cout<<" The array elements are: ";

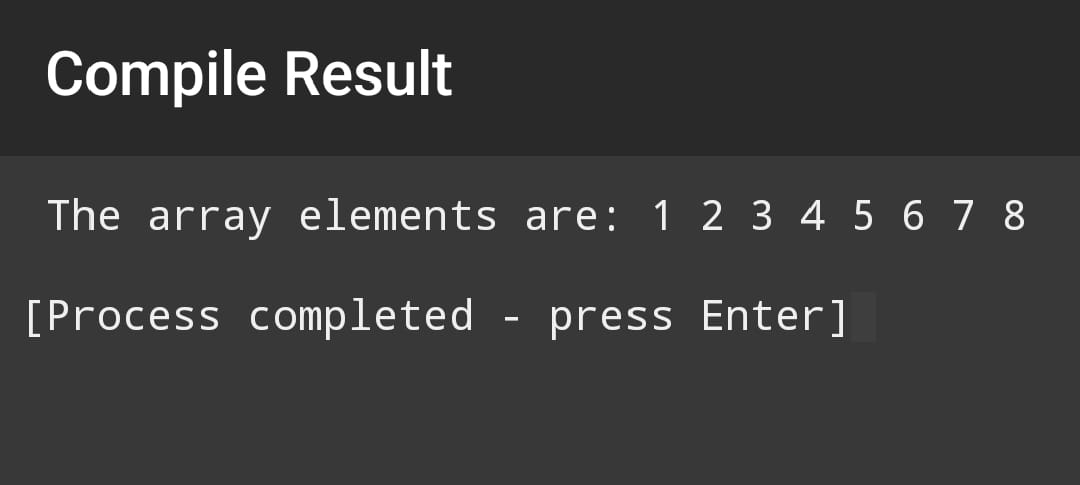
for( int i=0;i<newSize;++i)

{cout<<orgs[i]<<" ";}

cout<<endl;

return 0;

} \*/



/\* Q.09

Finding triplets\*/

/\* int main(){

int x,y,z,S;

int sum;

int arr[8];

cout<<" Enter 8 integers for array :"<<endl;

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

{ cin>>arr[i]; }

bool checked = false;

cout<<" Enter the sum integer value : ";

cin>>S;

cout<<" The required Triplets to sum are : ";

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

{

for( int j = 0; j<8; j++)

{

if(i==j)

continue;

for(int k = 0; k<8; k++)

{

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

continue;

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

if (sum==S) {

cout<<" { " <<arr[i]<< ", " <<arr[j]<<", " <<arr[k]<< " }";

checked = true;

}

}

}

}

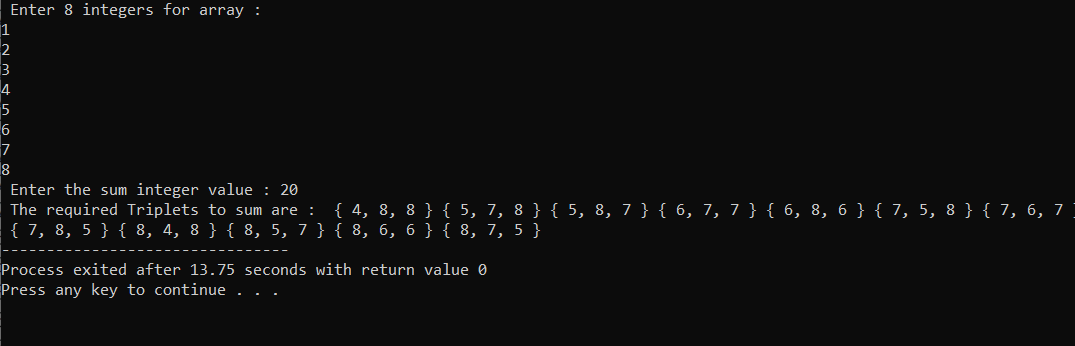
if (checked==false){

cout<< " There exists no sum."<<endl;

}

return 0;

} \*/



/\* Q. 10

Implement buble sort\*/

/\* int main()

{

int n;

cout << " Enter total no of elements : " ;

cin>> n;

int arr[n];

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

cout<<" The "<<i<<" space element in unsorted array is: ";

cin>>arr[i]; }

int counter = 1;

while ( counter<n-1){

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

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

int temp = arr[i];

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

arr[i+1] = temp; }

}

counter ++;

}

cout<<" The sorted array is : ";

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

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

cout<<endl;

return 0; } \*/

