******

***School of Mechanical & Manufacturing Engineering (SMME),***

***National University of Science and Technology (NUST),***

***Sector H-12, Islamabad***

Program: BE-Aerospace Section: AE-01

Session: Fall 2023 Semester: 1st

Course Title: Fundamentals of Programming (CS-109)

***“Assignment No.1”***

***Name:* Muhammad Daniyal Tariq**

***CMS:*** **463001**

***Question # 1***

**Write a C++ program, take two strings 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 code:***

**#include <iostream>**

**#include <string>**

**using namespace std;**

**int main() {**

**string str1st , str2nd ;**

**cout << "Write the 1st string: ";**

**cin >> str1st;**

**cout << "Write the 2nd string: ";**

**cin >> str2nd;**

**int l1 = str1st.length() , l2 = str2nd.length();**

**bool notequal = false;**

**char temp;**

**for (int i = 0; i < l1; i++) {**

**if (l1 != l2 && str1st != str2nd) {**

**notequal = true;**

**break;**

**}**

**}**

**if (notequal) {**

**cout << "String are not equal" << endl;**

**}else {**

**for (int i = 0, j = l1 - 1; i < j; i++, j--) {**

**temp = str1st[i];**

**str1st[i] = str1st[j];**

**str1st[j] = temp;**

**}**

**cout << "The rotated string is : " << str1st;**

**}**

**return 0;**

**}**

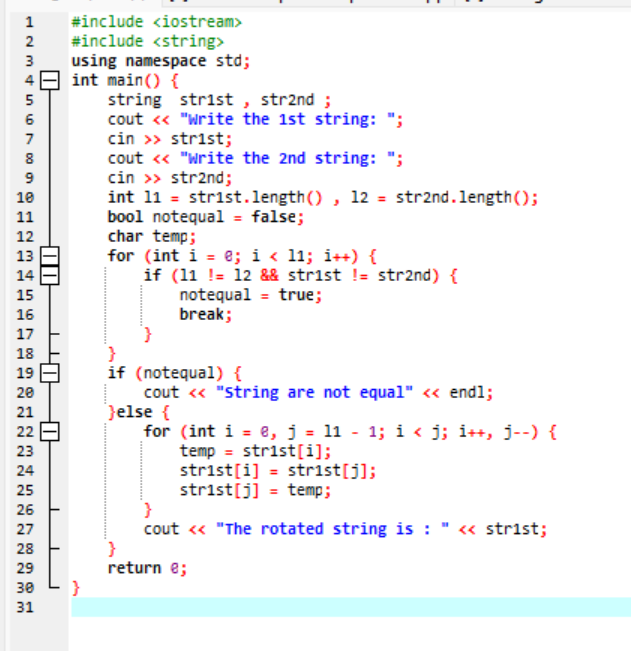
****

Figure 1: screen shot of code on Dev C++

***CPP File***



***Explanation***

I've updated the C++ code to prompt users for two strings and check if they're equal in both length and content. If they aren't equal, a message will be displayed informing the user. However, if the strings are equal, the first string will be rotated by reversing its characters and the rotated string will be displayed. The code has been improved to simplify the equality check and separate it from the string rotation logic, resulting in better efficiency and readability

***Output***

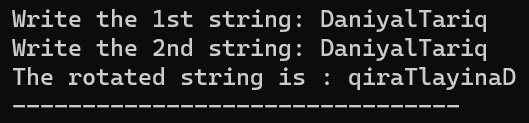
****

Figure 2: Output1 for Q#1

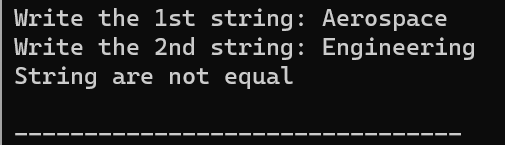
****

Figure 3:Output2 for Q#1

***Question # 2***

**Write a C++ program for a string containing lowercase and uppercase characters. The task is to remove all duplicate characters from the string and find the resultant string.**

***Solution Code:***

**#include <iostream>**

**#include <string>**

**using namespace std;**

**int main() {**

**string ing;**

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

**getline(cin, ing);**

**int l = ing.length();**

**cout << endl;**

**for (int i = 0; i < l; i++) {**

**bool diff = true;**

**for (int j = 0; j < l; j++) {**

**if (i != j && ing[i] == ing[j]){**

**diff = false;**

**break;**

**}**

**}**

**if (diff) {**

**cout << ing[i];**

**}**

**}**

**return 0;**

**}**

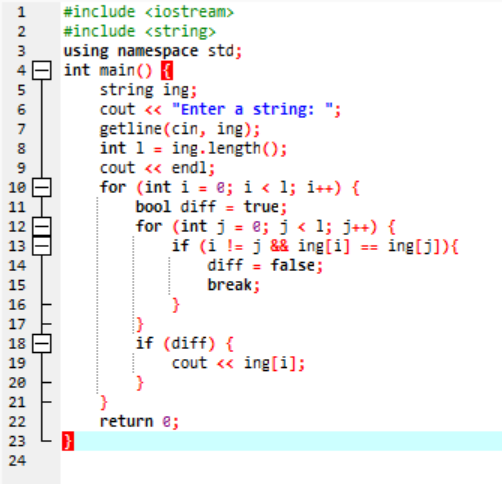
****

Figure 4: Screenshot of above given code on Dev C++

***CPP File***



***Explanation***

Here's a piece of code that allows users to input a string and returns only the unique characters in that string, while maintaining their original order. It does this by going through each character in the string and checking if it occurs anywhere else in the string, and then only displaying the characters that occur once. The main objective of this code is to identify and display all the unique characters present in the given string.

***Output***

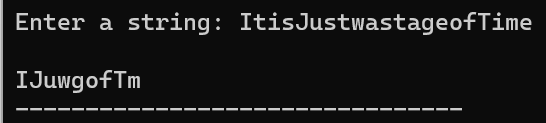


Figure 5: Output for Q#2

***Question # 3***

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

***Solution code:***

**#include <iostream>**

**using namespace std;**

**int main() {**

**int n = 5 , k ;**

**int arrA[n] = {1, 2, 3, 4, 5};**

**cout <<"Enter number of ellements 'N' you like to add in array."<<endl<<"N=";**

**cin>>k;**

**cout << "Type in the elements you would like to add: " << endl;**

**for (int i = n; i < n+k; i++) {**

**cin >> arrA[i];**

**}**

**cout<<"The array after element added :"<<endl<<"{";**

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

**cout << arrA[i]<<",";**

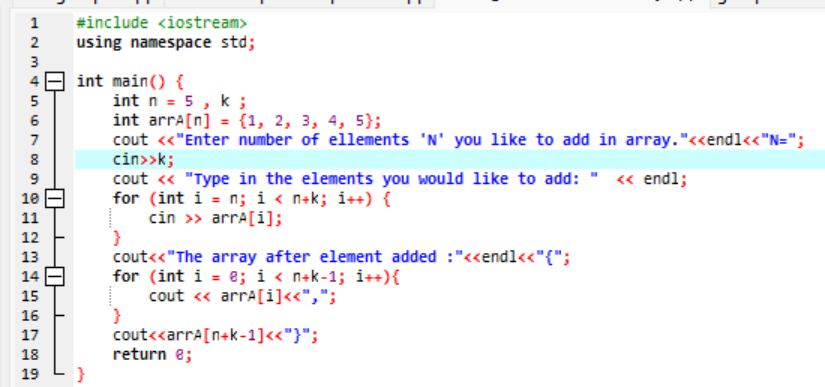
**}**

**cout<<arrA[n+k-1]<<"}";**

**return 0;**

**}**

Figure 6: Screenshot of above code on Dev C++

******

***CPP File***



***Explanation***

Here's an overview of a C++ program. The program initializes an array called arrA with five elements and asks the user to input the number of extra elements they want to add (k). The user is then asked to enter the values for the new elements, which are appended to the existing array. The updated array is then displayed. It's important to note that the program doesn't handle cases where invalid input may occur, so it assumes that the user will input a valid number for k. Another thing to note is that the final output includes a comma after each element, even the last one, which may result in a slightly incorrect array representation.

***Output***

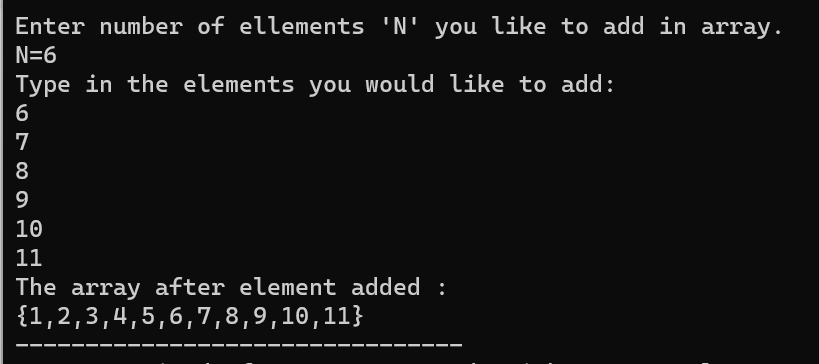
******

Figure 7: Output for Q#3

***Question # 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.**

***Solution Code:***

**#include <iostream>**

**using namespace std;**

**int main() {**

**int num;**

**cout << "Type any number 'N': ";**

**cin >> num;**

**if (num < 2) {**

**cout << "There is no Prime Number less than 2 as 2 is the smallest prime number!";**

**}**

**else if (num == 2) {**

**cout << "The largest prime number less than or equal to " <<num<< " is: 2";**

**}**

**else {**

**for (int i = num ; i >= 2; i--) {**

**bool PR = true;**

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

**if (i % j == 0) {**

**PR = false;**

**break;**

**}**

**}**

**if (PR) {**

**cout << "The largest prime number less than or equal to " <<num<< " is: " << i;**

**break;**

**}**

**}**

**}**

**return 0;**

**}**

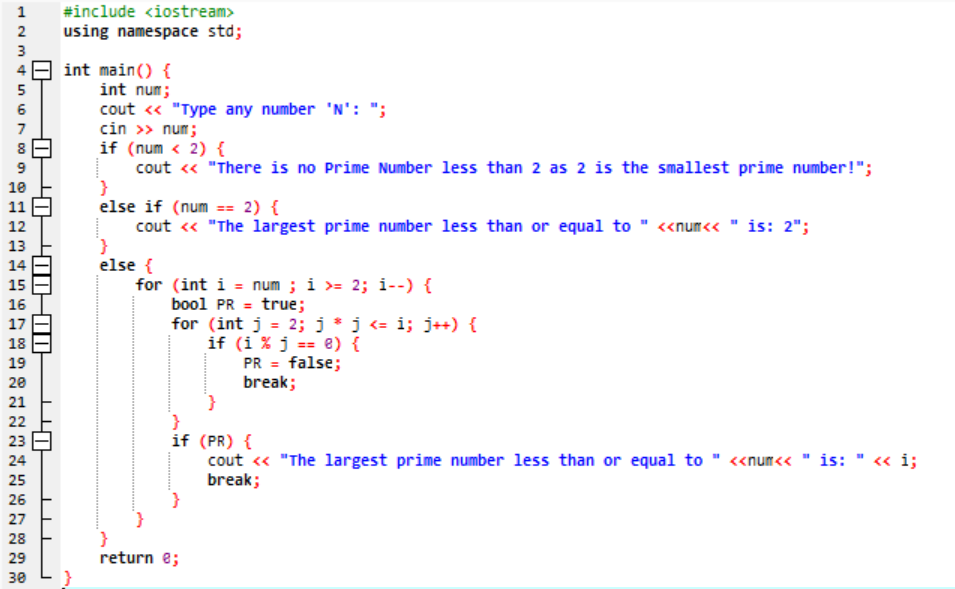
****

Figure 8: Screenshot for above written code Dev C++

***CPP File***



***Explanation***

The above written piece of code allows the user to enter a string and then displays the unique characters within that string, while retaining their original order. The code works by scanning each character in the string and verifying whether or not it appears elsewhere in the string. Only the characters that occur once are printed, resulting in a list of unique characters. Essentially, this code is designed to identify and output the distinct characters present in the input string.

***Output***

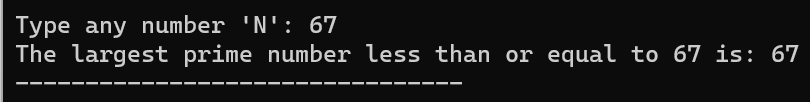
******

Figure 9: Output for Q#4

***Question # 5***

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

***Solution Code***

**#include <iostream>**

**using namespace std;**

**int main() {**

**int arr[6] = {29, 912, 30, 34, 19, 679};**

**int i, j, temp;**

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

**for (j = i; j < 6; j++) {**

**if (arr[i] < arr[j]) {**

**temp = arr[i];**

**arr[i] = arr[j];**

**arr[j] = temp;**

**}**

**}**

**}**

**cout<< "The array in descending order is:"<<endl;**

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

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

**}**

**return 0;**

**}**

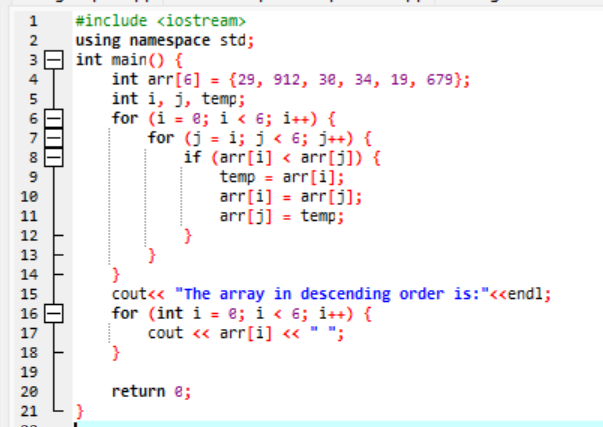
****

Figure 10: Screenshot of above written code in Dev C++

***CPP File***



***Explanation***

Here's an example of a C++ code that uses the Bubble Sort algorithm to sort an array of six integers in descending order. The algorithm compares adjacent elements and swaps them if they are in the wrong order, eventually resulting in a sorted array arranged from highest to lowest. The sorted array is then displayed as output.

***Output***

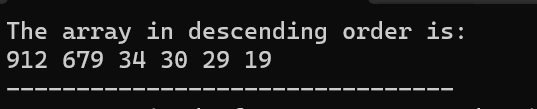
******

Figure 11: Output for Q#5

***Question # 5***

**Solve any Aerospace/Real Life Problem using C++ Programming.**

***Astronautical Problem***

Bernoulli's equation is very important equation aerodynamics or aerospace engineering that relates the static pressure, dynamic pressure, and potential energy per unit volume of a fluid. The equation is:

***P + 1/2v2  + = constant***

**Where**

* ‘***P***’ is the static pressure.
* ‘**’** is the density of the fluid.
* ‘***v***’ is the relative velocity of the fluid.
* ‘***g***’ is the gravitational acceleration.
* ‘***h***’ is the height of an object in the fluid concerning the earth's surface.

Write a C++ program by which we can we find the value of static pressure on an object at any moment

during its journey.

***Solution code:***

**#include <iostream>**

**#include <cmath>**

**using namespace std;**

**int main (){**

**double sp1, rho, h1, g=9.8, v1, k;**

**cout<<"Enter the values for initial conditions"<<endl<<"static pressure=";**

**cin>>sp1; cout<<"density="; cin>>rho; cout<<"height="; cin>>h1;**

**cout<<"relative velocity="; cin>>v1; cout <<endl;**

**k= sp1 + (rho\*pow(v1,2))/2 + rho\*g\*h1;**

**double sp2, h2, v2;**

**cout<<"Give the values of velocity and height at this moment:"<<endl;**

**cout<<"height="; cin>>h2; cout<<"relative velocity="; cin>>v2;**

**sp2= k- (rho\*pow(v2,2))/2 - rho\*g\*h2;**

**if (sp2<0){**

**sp2= -1\*sp2;**

**}**

**cout<< "Static pressure on the object = "<<sp2;**

**return 0;**

**}**

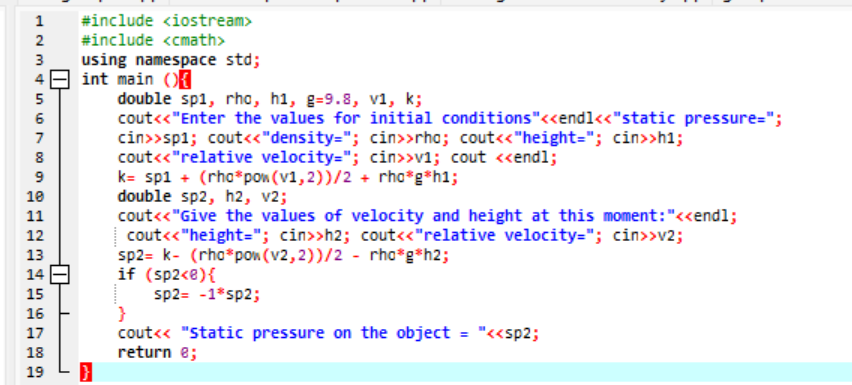
****

Figure 12: screenshot of above written code in Dev C++

***CPP File***



***Explanation***

To solve this aerospace problem, I write a very simple program in which I firstly take initial conditions(static pressure, velocity, height and density etc.) measure the constant of bernoulli’s equation. And then program will take height and relative velocity as input and measure the static pressure at any value of velocity or height .

***Output***

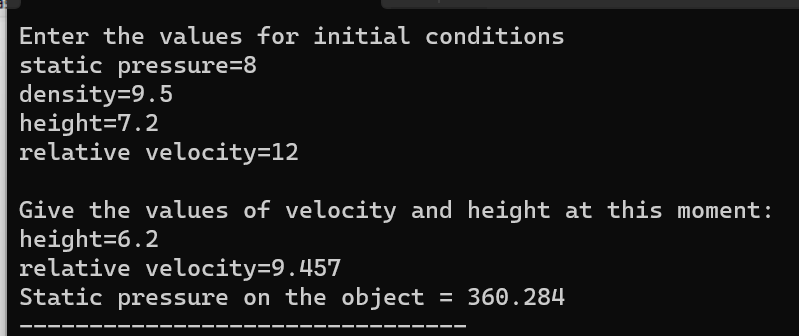
******

Figure 13: the output