School of Mechanical & Manufacturing Engineering (SMME),

National University of Science and Technology (NUST),

Sector H-12, Islamabad

Program:BE Aerospace Engineering Section:AE-01

Session:Fall 2023 Semester:1st

Course Title:Fundamentals of Programming,CS-109

Assignment # 1

Name : Aisha Iqbal

CMS : 456928

Question # 1:

Write a C++ program, take two strings as input from the 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.

Input:

```cpp
#include <iostream>
#include <string>
using namespace std;

string rotateString(string string) {
    return string.substr(1) + string[0];
}

int main() {
    string string1, string2;
    cout << "Enter first string: ";
    cin >> string1;
    cout << "Enter second string: ";
    cin >> string2;

    if (string1 == string2) {
        cout << "Strings are equal." << endl;
        string1 = rotateString(string1);
        string2 = rotateString(string2);
        cout << "Rotated strings: " << string1 << " and " << string2 << endl;
    } else {
        cout << "Strings are not equal." << endl;
    }

    return 0;
}
```

Key points:

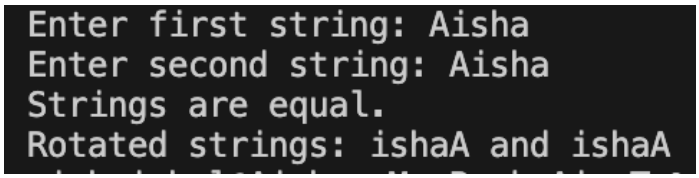
*A string can be rotated by one position to the left using the rotateString function.

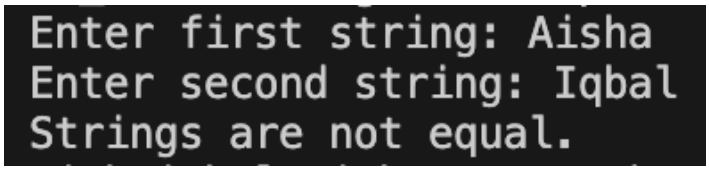
*The user is asked to enter two strings, string1 and string2, in main().

*It uses the == operator to determine whether the strings are equal.

*If both strings are equal, the rotateString function is used to rotate both strings and a message is printed.
*At last, the rotated strings are shown.

Output:





Question # 2:

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.

Input:

```cpp
#include <iostream>
#include <string>
using namespace std;

string removeDuplicates(const string& str) {
    string result;
    for (char ch : str) {
        if (result.find(tolower(ch)) == string::npos) {
            result += ch;
        }
    }
    return result;
}

int main() {
    string input;
    cout << "Enter a string with uppercase and lowercase characters: ";
    getline(cin, input);

    string result = removeDuplicates(input);

    cout << "String after removing duplicates: " << result << endl;

    return 0;
}
```

Key Points:

*A function called removeDuplicates is defined in the programme to eliminate duplicate characters from a string while taking case differences into account.

*The user is prompted to provide a string that contains both capital and lowercase characters by the main() method.

*Getline is used to read user input, which is then saved in the input string.

*To process the input and store the outcome in the result string, execute the removeDuplicates function.

*After removing any duplicate characters from the original string, the programme prints the amended string to the console.

Output:

```
Enter a string with uppercase and lowercase characters: my name is AISHA
String after removing duplicates: my naeisH
```

Question # 3:

Suppose an integer array1[5] = {1,2,3,4,5}. Add more elements to it and display them in C++.

Input:

```cpp
#include <iostream>
using namespace std;

int main() {
    int Array1[] = {1, 2, 3, 4, 5};
    int newSize = 8;
    int Array2[] = {6, 7, 8};

    int merged[newSize];

    copy(begin(Array1), end(Array1), merged);
    copy(begin(Array2), end(Array2), merged + sizeof(Array1) / sizeof(Array1[0]));

    cout << "Elements in the merged array: ";
    for (int i = 0; i < newSize; ++i) {
        std::cout << merged[i] << " ";
    }
    cout << endl;

    return 0;
}
```

Key Points:

*Two arrays, Array1 and Array2, are initialized and newSize is defined as 8.
*It produces an array with merged size newSize.
*Attempts to use copy to combine the arrays; however, there may be a mistake in the computation of where to begin copying Array2.
*It employs a for loop to produce the elements in the combined array.
*Array2 may not have been merged correctly by the merging procedure, which could produce inaccurate results.

Output:

```
Elements in the merged array: 1 2 3 4 5 6 7 8
```

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.

Input:

```cpp
#include <iostream>
using namespace std;
int main() {
    int N;
    cout << "Enter a number: ";
    cin >> N;
    while (N > 1) {
        bool flag = true;
        for (int i = 2; i * i <= N; ++i) {
            if (N % i == 0) {
                flag = false;
                break;
            }
        }
        if (flag == true) {
            cout << "Largest prime number less than or equal to N: " << N << endl;
            return 0;
        }

        N--;
    }
    cout << "No prime number found less than number" << endl;
    return 0;
}
```

Key points:

*The code asks the user to enter the number N.

*Then it employs a loop to decrease N and iteratively goes from 2 to the square root of N to determine if each value is primitive.

*When a prime number is identified, the programme ends and prints the greatest prime number that is less than or equal to N.

*After the loop, if no prime number is found, a notice stating that no prime number less than the input was found is displayed.

*By iterating backwards, this method finds the highest prime number that is less than or equal to the input by testing each number up to 1.

Output:

```
Enter a number: 15
Largest prime number less than or equal to N: 13
```

Question # 5:

Implement Bubble Sort on an array of 6 integers.

Input:

```cpp
#include <iostream>
using namespace std;
int main() {
    int temp;
    int a[6] = {60,47,96, 11, 4, 2};
    for (int i = 0 ; i < 6; i++){
        for (int j = 0; j < 5; j++){
            if (a[j] > a[j+1]){
                temp = a[j];
                a[j] = a[j+1];
                a[j+1] = temp;
            }
        }
    }
    for (int k=0; k < 6; k++){
        cout << a[k] << "  ";
    }
    return 0;
}
```

Key points:

*The programme applies the Bubble Sort algorithm after initializing an array with integers.

*In nested for loops, adjacent elements are compared as they run across the array.

*Up until the array is sorted, it switches elements if one is bigger than the next.

*Print the array's elements in ascending order after sorting.

*In order to fully sort an array, Bubble Sort iteratively compares adjacent elements, progressively pushing the larger elements towards the end.

Output:

```
 2  4  11  47  60  96
```

Question # 6:

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

Input:

This C++ program enables the computation of an aircraft's endurance based on user-input fuel capacity and consumption rate.

```cpp
#include <iostream>
using namespace std;

double calculateEndurance(double fuelCapacity, double fuelConsumptionRate) {
    double endurance = fuelCapacity / fuelConsumptionRate;
    return endurance;
}

int main() {
    double fuelCapacity, fuelConsumptionRate;

    cout << "Enter the fuel capacity of the aircraft (in liters): ";
    cin >> fuelCapacity;

    cout << "Enter the fuel consumption rate of the aircraft (in liters per hour): ";
    cin >> fuelConsumptionRate;

    double endurance = calculateEndurance(fuelCapacity, fuelConsumptionRate);

    cout << "Endurance: " << endurance << " hours\n";

    return 0;
}
```

Key points:

*Input Collection: Compiles the aircraft's user-specified fuel capacity (measured in liters) and consumption rate (measured in liters per hour).

*Endurance Calculation: By dividing fuel capacity by consumption rate, a separate function is used to determine the aircraft's endurance.

*Function Modularity: By containing the endurance calculation inside a specific function, this shows modularity.

*Calculation Output: Shows the user the endurance value that was calculated in hours.

Output:

```
Enter the fuel capacity of the aircraft (in liters): 2000
Enter the fuel consumption rate of the aircraft (in liters per hour): 200
Endurance: 10 hours
```