

1- Write a program to determine if two arrays A and B of equal size N are equal or not. Two arrays are considered equal if they contain the same set of elements, although the arrangements (or permutations) of the elements may be different. If there are repeated elements, the counts of repeated elements must also be the same for the two arrays to be considered equal.

اكتب برنامجًا لتحديد ما إذا كانت المصفوفة A و B اللتان لهما نفس الحجم N متساويتين أم لا. يعتبر Two arrays متساويتين إذا كانا يحتويان على نفس مجموعة العناصر، على الرغم من أن الترتيبات (أو التباديل) للعناصر قد تكون مختلفة. إذا كانت هناك عناصر متكررة، فيجب أيضًا أن يكون عدد العناصر المتكررة هو نفسه حتى يتم اعتبار Two arrays متساويتين.

Output

```
Output: 1
```

Solution

```
// www.gammal.tech

#include <iostream>
#include <unordered_map>
#include <vector>
using namespace std;

bool areEqualArrays(int A[], int B[], int N) {
    unordered_map<int, int> freqA, freqB;

    // Count frequency of elements in array A
    for (int i = 0; i < N; ++i)
        freqA[A[i]]++;

    // Count frequency of elements in array B
    for (int i = 0; i < N; ++i)
        freqB[B[i]]++;

    // Check if the frequency maps are equal
    return freqA == freqB;
}

int main() {
    int N = 5;
    int A[] = {1, 2, 5, 4, 0};
    int B[] = {2, 4, 5, 0, 1};

    if (areEqualArrays(A, B, N))
        cout << "Output: 1\n";
    else
        cout << "Output: 0\n";

    return 0;
}
```

2- Write a program that prompts the user to enter a list of names and then displays the longest name from the entered list. If there are multiple names of the longest size, the program should return the first occurring name.

كتابة برنامج يطلب من المستخدم إدخال قائمة الأسماء ثم يعرض الاسم الأطول من القائمة المدخلة. إذا كان هناك أسماء متعددة ذات الحجم الأطول، فيجب أن يقوم البرنامج بإرجاع الاسم الأول.

Input

```
Enter the number of names: 3
Enter 3 names:
aly
ahmed
amr
```

Output

```
Longest name: ahmed
```

Solution

```
// www.gammal.tech

#include <iostream>
#include <vector>
#include <string>
using namespace std;

string findLongestName(const vector<string>& names) {
    string longestName = "";
    for (const string& name : names) {
        if (name.size() > longestName.size()) {
            longestName = name;
        }
    }
    return longestName;
}

int main() {
    // Prompt the user to enter the number of names
    int numNames;
    cout << "Enter the number of names: ";
    cin >> numNames;

    // Prompt the user to enter the names
    vector<string> names;
    cout << "Enter " << numNames << " names:" << endl;
    for (int i = 0; i < numNames; ++i) {
        string name;
        cin >> name;
        names.push_back(name);
    }

    // Find the longest name
    string longestName = findLongestName(names);

    // Output the result
    cout << "Longest name: " << longestName << endl;

    return 0;
}
```

3- Write a program to determine if a given non-empty sequence of characters is binary. Return true if the sequence is binary, otherwise return false. A sequence is considered binary if it contains only '0's and '1's.

اكتب برنامجًا لتحديد ما إذا كان تسلسل الأحرف غير الفارغ معينًا ثنائيًا. يُرجع صحيحًا إذا كان التسلسل ثنائيًا، وإلا يُرجع خطأ. يعتبر التسلسل ثنائيًا إذا كان يحتوي على "0" و"1" فقط.

Input

```
Enter a sequence of characters: 10101
```

Output

```
The sequence is binary.
```

Solution

```
// www.gammal.tech

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

bool isBinary(const string& str) {
    for (char c : str) {
        if (c != '0' && c != '1') {
            return false;
        }
    }
    return true;
}

int main() {
    string sequence;
    cout << "Enter a sequence of characters: ";
    cin >> sequence;

    if (isBinary(sequence)) {
        cout << "The sequence is binary." << endl;
    } else {
        cout << "The sequence is not binary." << endl;
    }

    return 0;
}
```

4- Write a program to check if all characters in a given string are the same or not. Return true if all characters are the same, otherwise return false.

اكتب برنامجًا للتحقق مما إذا كانت جميع الأحرف في سلسلة معينة متماثلة أم لا.
يُرجع صحيحًا إذا كانت جميع الأحرف متماثلة، وإلا يُعاد خطأ.

Input

```
Enter a string: 111111
```

Output

```
All characters in the string are the same.
```

Solution

```
// www.gammal.tech

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

bool allCharactersSame(const string& str) {
    char firstChar = str[0];
    for (char c : str) {
        if (c != firstChar) {
            return false;
        }
    }
    return true;
}

int main() {
    string input;
    cout << "Enter a string: ";
    cin >> input;

    if (allCharactersSame(input)) {
        cout << "All characters in the string are the same." << endl;
    } else {
        cout << "Not all characters in the string are the same." << endl;
    }

    return 0;
}
```

5- Write a program to count the occurrence of lowercase characters, uppercase characters, special characters, and numeric values in a given string.

اكتب برنامجًا لحساب عدد الأحرف الصغيرة والأحرف الكبيرة والأحرف الخاصة والقيم الرقمية في سلسلة معينة.

Input

```
Enter a string: Hello! How are You? 1234
```

Output

```
Lowercase characters: 11
Uppercase characters: 3
Numeric values: 4
Special characters: 6
```

Solution

```
// www.gammal.tech

#include <iostream>
#include <string>
#include <cctype>
using namespace std;

void countCharacters(const string& str) {
    int lowercaseCount = 0, uppercaseCount = 0, specialCount = 0, numericCount = 0;

    for (char c : str) {
        if (islower(c)) {
            lowercaseCount++;
        } else if (isupper(c)) {
            uppercaseCount++;
        } else if (isdigit(c)) {
            numericCount++;
        } else {
            specialCount++;
        }
    }

    cout << "Lowercase characters: " << lowercaseCount << endl;
    cout << "Uppercase characters: " << uppercaseCount << endl;
    cout << "Numeric values: " << numericCount << endl;
    cout << "Special characters: " << specialCount << endl;
}

int main() {
    string input;
    cout << "Enter a string: ";
    getline(cin, input);

    countCharacters(input);

    return 0;
}
```

6- Write a program that takes a string containing only lowercase letters and generates a new string with the same letters but in uppercase.

اكتب برنامجًا يأخذ سلسلة تحتوي على أحرف صغيرة فقط وينشئ سلسلة جديدة بنفس الأحرف ولكن بأحرف كبيرة.

Input

```
Enter a string in lowercase: hello
```

Output

```
String in uppercase: HELLO
```

Solution

```
// www.gammal.tech

#include <iostream>
#include <string>
#include <cctype>
using namespace std;

string convertToUppercase(const string& str) {
    string result = str;

    for (char& c : result) {
        if (islower(c)) {
            c = toupper(c);
        }
    }

    return result;
}

int main() {
    string input;
    cout << "Enter a string in lowercase: ";
    cin >> input;

    string result = convertToUppercase(input);
    cout << "String in uppercase: " << result << endl;

    return 0;
}
```

7- Write a program to remove spaces from a given string and print the resulting string.

اكتب برنامجًا لإزالة المسافات من سلسلة معينة وطباعة السلسلة الناتجة.

Input

```
Enter a string with spaces: Hello World
```

Output

```
String after removing spaces: HelloWorld
```

Solution

```
// www.gammal.tech

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

string removeSpaces(string str) {
    string result = "";
    for (char c : str) {
        if (c != ' ') {
            result += c;
        }
    }
    return result;
}

int main() {
    string input;
    cout << "Enter a string with spaces: ";
    getline(cin, input);

    string result = removeSpaces(input);
    cout << "String after removing spaces: " << result << endl;

    return 0;
}
```


8- Write a program that takes a string str consisting of lowercase alphabets. Your task is to find the maximum occurring character in the string str. If more than one character occurs the maximum number of times, then print the lexicographically smallest character among them.

كتابة برنامج يأخذ سلسلة تتكون من الحروف الهجائية الصغيرة. مهمتك هي العثور على الحد الأقصى لعدد الأحرف في السلسلة. في حالة تكرار أكثر من حرف الحد الأقصى لعدد المرات، قم بطباعة أصغر حرف من بينها من الناحية المعجمية.

Input

```
Enter the string: Hello
```

Output

```
Maximum occurring character: l
```

Solution

```
// www.gammal.tech
#include <iostream>
#include <unordered_map>
using namespace std;

char maxOccurringCharacter(string str) {
    unordered_map<char, int> frequency;

    // Count the frequency of each character in the string
    for (char c : str) {
        frequency[c]++;
    }

    char maxChar = 'a'; // Initialize maxChar with any lowercase alphabet
    int maxCount = 0;   // Initialize maxCount

    // Iterate through the frequency map to find the maximum occurring character
    for (auto& pair : frequency) {
        if (pair.second > maxCount || (pair.second == maxCount && pair.first < maxChar)) {
            maxChar = pair.first;
            maxCount = pair.second;
        }
    }

    return maxChar;
}

int main() {
    string str;
    cout << "Enter the string: ";
    cin >> str;

    char result = maxOccurringCharacter(str);
    cout << "Maximum occurring character: " << result << endl;

    return 0;
}
```

9- Write a program where you are given a string containing alphanumeric characters. Your task is to find the sum of all the numbers present in the string.

اكتب برنامجًا يُعطى لك سلسلة تحتوي على أحرف أبجدية رقمية. مهمتك هي العثور على مجموع كل الأرقام الموجودة في السلسلة.

Input

```
Enter the string: he45l5
```

Output

```
Sum of numbers: 50
```

Solution

```
// www.gammal.tech

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

int sumOfNumbersInString(string str) {
    int sum = 0;
    int num = 0;

    // Iterate through the characters in the string
    for (char c : str) {
        if (isdigit(c)) {
            // If the character is a digit, update the current number
            num = num * 10 + (c - '0');
        } else {
            // If the character is not a digit, add the current number to the sum
            sum += num;
            num = 0; // Reset the current number
        }
    }

    // Add the last number to the sum
    sum += num;

    return sum;
}

int main() {
    string str;
    cout << "Enter the string: ";
    cin >> str;

    int result = sumOfNumbersInString(str);
    cout << "Sum of numbers: " << result << endl;

    return 0;
}
```

10- Write a program where You are given a string S consisting of upper/lower-case alphabets and empty space characters ' '. The string may contain spaces at the end. Your task is to return the length of the last word in the string which consists of alphabets only.

اكتب برنامجًا حيث يتم إعطاؤك سلسلة S تتكون من حروف أبجدية كبيرة/صغيرة ومسافات فارغة ' '. قد تحتوي السلسلة على مسافات في النهاية. مهمتك هي إرجاع طول الكلمة الأخيرة في السلسلة التي تتكون من الحروف الهجائية فقط.

Input

```
Enter the string: Hello Gammal Tech
```

Output

```
Length of last word: 4
```

Solution

```
// www.gammal.tech

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

int lengthOfLastWord(string s) {
    int length = 0;
    int i = s.length() - 1;

    // Skip trailing spaces
    while (i >= 0 && s[i] == ' ') {
        i--;
    }

    // Calculate length of last word
    while (i >= 0 && s[i] != ' ') {
        length++;
        i--;
    }

    return length;
}

int main() {
    string s;
    cout << "Enter the string: ";
    getline(cin, s);

    int result = lengthOfLastWord(s);
    cout << "Length of last word: " << result << endl;

    return 0;
}
```