A blue and white logo

Description automatically generated

**Assignment 2**

Submitted By M Shahwaiz Shahid

Reg no. FA22 BCS 049

Subject DSA

Submitted To Ms. Tahreem Saeed

Date 25th Oct 2023

**Question 1:**

**Write down C/C++ codes for the following functions of strings data structure:**

**a.Removing punctuations from a given string**

**Program:**

#include <iostream>

#include <string>

#include <cctype>

using namespace std;

string removePunctuations(const std::string& input) {

string result = "";

for (char c : input) {

if (!ispunct(c)) {

result += c;

}

}

return result;

}

int main() {

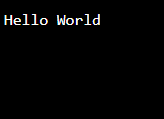
string input = "Hello, World!";

string result = removePunctuations(input);

cout << result << std::endl;

return 0;

}



**b. Rearrange characters in a string such that no two adjacent are same.**

**Program:**

#include <iostream>

#include <string>

#include <algorithm>

using namespace std;

string rearrangeCharacters(const std::string& input) {

string str = input;

if (str.size() <= 1) return str;

for (int i = 1; i < str.size(); i += 2) {

if (str[i] == str[i - 1]) {

for (int j = i + 1; j < str.size(); j++) {

if (str[j] != str[i - 1]) {

swap(str[i], str[j]);

break;

}

}

}

}

return str;

}

int main() {

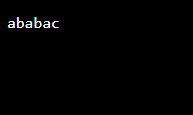
string input = "aaabbc";

string result = rearrangeCharacters(input);

cout << result << std::endl;

return 0;

}



**c. Program to check if input is an integer or a string**

**Program**

#include <iostream>

#include <string>

#include <cctype>

using namespace std;

bool isInteger(const string &input) {

for (char c : input) {

if (!isdigit(c)) {

return false;

}

}

return true;

}

int main() {

string input = "12345";

if (isInteger(input)) {

cout << "Input is an integer." <<endl;

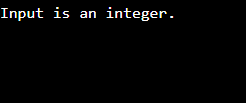
} else {

cout << "Input is not an integer." <<endl;

}

return 0;

}



**d. C++ program to find second most frequent character.**

**Program**

#include <iostream>

#include <string>

#include <map>

using namespace std;

char findSecondMostFrequentChar(const string& str) {

map<char, int> charCount;

for (char c : str) {

if (isalpha(c)) {

charCount[c]++;

}

}

char mostFrequent = ' ';

int maxCount = 0;

char secondMostFrequent = ' ';

int secondMaxCount = 0;

for (const auto& pair : charCount) {

if (pair.second > maxCount) {

secondMostFrequent = mostFrequent;

secondMaxCount = maxCount;

mostFrequent = pair.first;

maxCount = pair.second;

} else if (pair.second > secondMaxCount) {

secondMostFrequent = pair.first;

secondMaxCount = pair.second;

}

}

return secondMostFrequent;

}

int main() {

string input;

cout << "Enter a string: ";

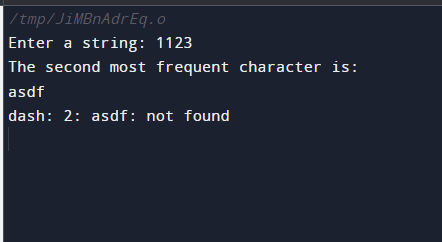
cin >> input;

char secondMost = findSecondMostFrequentChar(input);

cout << "The second most frequent character is: " << secondMost <<endl;

return 0;

}



**e. C++ Program to Sort an array of names or strings.**

**Program**

#include <iostream>

#include <string>

#include <vector>

#include <algorithm>

using namespace std;

int main() {

vector<string> names;

names.push\_back("Ali");

names.push\_back("Zain");

names.push\_back("Shahwaiz");

sort(names.begin(), names.end());

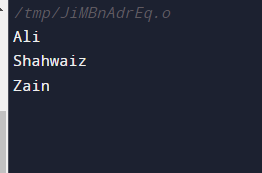
for (const string& name : names) {

cout << name << endl;

}

return 0;

}



**f. C++ program to concatenate a string given number of times.**

**Program**

#include <iostream>

#include <string>

using namespace std;

string concatenateString(const string& str, int times) {

string result;

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

result += str;

}

return result;

}

int main() {

string input;

int times;

cout << "Enter a string: ";

cin >> input;

cout << "Enter the number of times to concatenate: ";

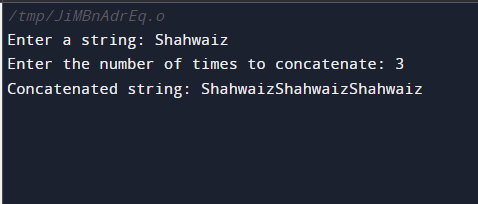
cin >> times;

string concatenated = concatenateString(input, times);

cout << "Concatenated string: " << concatenated << stdendl;

return 0;

}



**g. std::string::append vs std::string::push\_back() vs Operator += in C++.**

**program**

#include <iostream>

#include <string>

int main() {

std::string str1 = "Hello, ";

std::string str2 = "world!";

char ch = 'X';

str1.append(str2);

std::cout << "Using append: " << str1 << std::endl;

for (char c : str2) {

str1.push\_back(c);

}

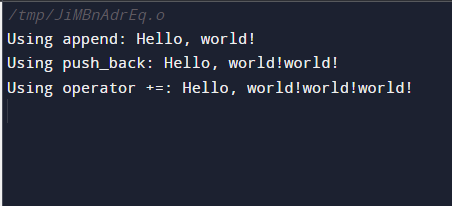
std::cout << "Using push\_back: " << str1 << std::endl;

str1 += str2;

std::cout << "Using operator +=: " << str1 << std::endl;

return 0;

}



**h. Comparing two strings in C++.**

**Program**

#include <iostream>

#include <string>

using namespace std;

int main() {

string str1 = "apple";

string str2 = "banana";

if (str1 == str2) {

cout << "Strings are equal." << std::endl;

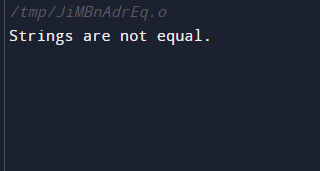
} else {

cout << "Strings are not equal." << std::endl;

}

return 0;

}



**I.Extract all integers from string in C++.**

**Program**

#include <iostream>

#include <string>

#include <cctype>

using namespace std;

int main() {

string input = "Hello 123 World 4567";

string number = "";

for (char c : input) {

if (isdigit(c)) {

number += c;

} else if (!number.empty()) {

cout << "Found number: " << number <<endl;

number.clear();

}

}

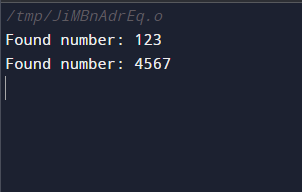
if (!number.empty()) {

cout << "Found number: " << number << std::endl;

}

return 0;

}



**j. C++ program to Replace a word in a text by another given word.**

**Program**

#include <iostream>

#include <string>

using namespace std;

int main() {

string text;

string targetWord, replacementWord;

cout << "Enter the text: ";

getline(cin, text);

cout << "Enter the word to replace: ";

cin >> targetWord;

cout << "Enter the replacement word: ";

cin >> replacementWord;

size\_t pos = text.find(targetWord);

if (pos !=string::npos) {

text.replace(pos, targetWord.length(), replacementWord);

cout << "Replaced text: " << text <<endl;

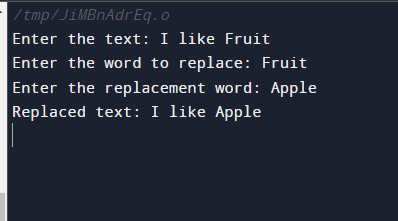
} else {

cout << "Word not found in the text." << endl;

}

return 0;

}



**………………………………………………………………**