Tabish Ahmad – 23103145

Assignment 4 – Strings

Question 1)

```
#include<iostream>
using namespace std;
int main(){
   int count = 0;
   string s = "ndajksnd12";    while(s[count] != '\0'){
      count++;}
   cout<<count;
   return 0;}
```

```
10

=== Code Execution Successful ===
```

Question 2)

```
#include<iostream>
using namespace std;
int main(){
   string s1 = "ndasl";
   string s2 = "21dsa";
   int lenght = 0;     while(s1[lenght] != '\0'){
      s2[lenght] = s1[lenght] ;
      lenght++;}
   cout<<s2;
   return 0;}
```

```
ndasl

=== Code Execution Successful ===
```

Question 3)

```
#include<iostream>
using namespace std;
int main(){
   string s1 = "Helo";
   string s2 = "dlrow";
   int lenght = 0, count = 0;
   while(s1[lenght] != '\0'){
      lenght++;}
   while(s2[count] != '\0'){
      s1+= s2[count];
      count++;}
   cout<<s1;
   return 0;}
```

```
Helodlrow

=== Code Execution Successful ===
```

Question 4)

```
#include<iostream>
using namespace std;
int main(){
   string s1 = "ndksalnd";
   string s2 = "ndksalnd";
   int len = 0, flag = 0;
   while(s1[len] != '\0'){
      if(s1[len] == s2[len]){
         flag++;
         len++;}}
   if(flag == s1.size()){
```

```
Both strings are the same.

=== Code Execution Successful ===
```

```
    cout<<"Both strings are the same.";
  }else{
    cout<<"Strings aren't the same.";}
  return 0;}
```

Question 5)

```
#include<iostream>
using namespace std;
int main(){
  string s = "'niggas";
  int lenght = 0;
  while(s[lenght] != '\0'){       lenght++;}
  int start = 0, end = lenght - 1;
  while(start < end){
    int temp = s[start];
    s[start] = s[end];
    s[end] = temp;
    end--;
    start++;}
  cout<<s;}
```

```
saggin'

=== Code Execution Successful ===
```

Question 6)

```
#include<iostream>
#include<string>
using namespace std;
int main() {
  string names[3][3] = {
    {"Raj", "Sumit", "Vivek"},
    {"Priya", "Vikram", "Tarun"},
    {"Rohit", "Meena", "Angel"}};
  string target;
  cout << "Enter the name to search: ";    cin >> target;
  int found = 0;
  for(int i = 0; i < 3; i++) {
    for(int j = 0; j < 3; j++) {
      if(names[i][j] == target) {
        cout << "Name found at position: [" << i << "][" << j << "]" << endl;
        found = 1;
        break;}}}
  if(!found) {
    cout << "Name not found in the matrix." << endl;}
  return 0;}
```

```
Enter the name to search: Meena
Name found at position: [2][1]


=== Code Execution Successful ===
```

Question 7)

```
#include<iostream>
using namespace std;
int main() {
  string str;
  cout << "Enter a string: ";
  cin >> str;
  int len = 0;
  while (str[len] != '\0') {
    len++;}
  int start = 0, end = len - 1;
  int isPalindrome = 1;
  while (start < end) {
    if (str[start] != str[end]) {
      isPalindrome = 0;
```

```
Enter a string: malayalam
The string is a palindrome.
```

```
        break;}
    start++;
    end--;}
  if (isPalindrome) {
    cout << "The string is a palindrome." << endl;
  } else {
    cout << "The string is not a palindrome." << endl;}
  return 0;}
```

Question 8)

```
#include<iostream>
using namespace std;
int main() {
  string str;
  cout << "Enter a string: ";
  cin >> str;
  int freq[256] = {0};
  int len = 0;
  while (str[len] != '\0') {
    freq[str[len]]++;
    len++;}
  char maxFreqChar = str[0];
  int maxFreq = freq[str[0]];
  for (int i = 1; i < len; i++) {
    if (freq[str[i]] > maxFreq) {
      maxFreq = freq[str[i]];
      maxFreqChar = str[i];}}
  cout << "The highest frequency character is: " << maxFreqChar << endl;
  cout << "Frequency: " << maxFreq << endl;
  return 0;}
```

```
Enter a string: yyyykbjohidpajdnksdayyyyyyyyyy
The highest frequency character is: y
Frequency: 14
```

Question 9)

```
#include <iostream>
using namespace std;
int main() {
  string str;
  cout << "Enter a string: ";
  cin >> str;
  for (int i = 0; i < str.length(); i++) {
    if (str[i] >= 'a' && str[i] <= 'z') {
      str[i] = str[i] - ('a' - 'A');}}
  cout << "Uppercase string: " << str << endl;
  return 0;}
```

```
Enter a string: 1ed21w2
Uppercase string: 1ED21W2
```

Question 10)

```
#include <iostream>
using namespace std;
int main() {
  string str1, str2;
  cout << "Enter the first string: ";
  cin >> str1;
  cout << "Enter the second string: ";
  cin >> str2;
  int len1 = 0, len2 = 0;
  while (str1[len1] != '\0') {
    len1++;}
```

```
  while (str2[len2] != '\0') {
    len2++;}
  bool found = false;
  for (int i = 0; i <= len1 - len2; i++) {
    int j;       for (j = 0; j < len2; j++) {
      if (str1[i + j] != str2[j]) {
        break;}}
    if (j == len2) {
      found = true;
      break;}}
  if (found) {
    cout << "The second string is a substring of the first string." << endl;
  } else {
    cout << "The second string is not a substring of the first string." << endl;}
  return 0;}
```

```
Enter the first string: tabishahamad
Enter the second string: bish
The second string is a substring of the first string.
```

Question 11)

```
#include <iostream>
using namespace std;
int main() {
  string sentence;
  int wordCount = 0;
  cout << "Enter a sentence: ";
  getline(cin, sentence);    for (int i = 0; i <
sentence.length(); i++) {
    if (sentence[i] == ' ') {
      wordCount++;}}
  if (sentence.length() > 0) {
    wordCount++;}
  cout << "Number of words in the sentence: " << wordCount << endl;
  return 0;}
```

```
Enter a sentence: How are you doing
Number of words in the sentence: 4


=== Code Execution Successful ===
```

Question 12)

```
#include <iostream>
using namespace std;
int main() {
  string sentence;
  cout << "Enter a sentence: ";
  getline(cin, sentence);
  for (int i = 0; i < sentence.length(); i++) {
    if (sentence[i] == ' ') {
      cout << endl;       } else {
      cout << sentence[i];}}
  return 0;}
```

```
Enter a sentence: I am doing good
I
am
doing
good
```

Question 13)

```
#include <iostream>
using namespace std;
int main() {
  string input;
  cout << "Enter a string: ";
  cin >> input;
  int frequency[256] = {0};
  for (int i = 0; i < input.length(); i++) {
    frequency[input[i]]++;}
  cout << "Character frequencies:\n";
```

```
Enter a string: TabishAhmad
Character frequencies:
A: 1
T: 1
a: 2
b: 1
d: 1
h: 2
i: 1
m: 1
s: 1
```

```
  for (int i = 0; i < 256; i++) {
    if (frequency[i] > 0) {
      cout << char(i) << ": " << frequency[i] << endl;}}
  return 0;}
```

Question 14)

```
#include <iostream>
using namespace std;
int main() {
  string sentence, word, longestWord;
  int maxLength = 0, wordLength = 0;
  cout << "Enter a sentence: ";
  getline(cin, sentence);
  sentence += ' ';
  for (char ch : sentence) {
    if (ch != ' ') {
      word += ch;          wordLength++;
    } else {
      if (wordLength > maxLength) {
        maxLength = wordLength;
        longestWord = word;}
      word = "";
      wordLength = 0;}}
  cout << "Longest word: " << longestWord << endl;
  cout << "Length: " << maxLength << endl;
  return 0;}
```

```
Enter a sentence: DSA is boring
Longest word: boring
Length: 6


=== Code Execution Successful ===
```

Question 15)

```
#include <iostream>
using namespace std;
int main() {
  string inputString;
  char charToReplace, replacementChar;
  cout << "Enter a string: ";
  getline(cin, inputString);
  cout << "Enter the character to replace: ";
  cin >> charToReplace;
  cout << "Enter the replacement character: ";
  cin >> replacementChar;
  for (int i = 0; i < inputString.length(); i++) {      if (inputString[i] == charToReplace) {
      inputString[i] = replacementChar;}}
  cout << "Modified string: " << inputString << endl;
  return 0;}
```

```
Enter a string: Tabish
Enter the character to replace: s
Enter the replacement character: c
Modified string: Tabich
```

Question 16)

```
#include <iostream>
using namespace std;
int main() {
  string inputString;
  cout << "Enter a string: ";
  getline(cin, inputString);
  for (int i = 0; i < inputString.length() - 1; i++) {
    for (int j = i + 1; j < inputString.length(); j++) {
      if (inputString[i] > inputString[j]) {
        char temp = inputString[i];
```

```
            inputString[i] = inputString[j];
            inputString[j] = temp;}}}
   cout << "Sorted string: " << inputString << endl;
   return 0;}
```

Question 17)

```
#include<iostream>
#include<string>
using namespace std;
int main() {
   string str;
   cout << "Enter a string: ";
   cin >> str;
   int freq[256] = {0};
   for (int i = 0; i < str.length(); i++) {
      freq[str[i]]++;}
   cout << "Characters without duplicates: ";
   for (int i = 0; i < str.length(); i++) {
      if (freq[str[i]] > 0) {
         cout << str[i];
         freq[str[i]] = 0;}}
   return 0;}
```

```
Enter a string: tabish ahmad
Characters without duplicates: tabish
```

Question 18)

```
#include<iostream>
#include<string>
using namespace std;
void countChars(const string& str, int count[]) {
   for (char ch : str) {
      count[ch - 'a']++;}}
bool areAnagrams(const string& str1, const string& str2) {
   if (str1.length() != str2.length()) {
      return false;}
   int count1[26] = {0};
   int count2[26] = {0};
   countChars(str1, count1);
   countChars(str2, count2);
   for (int i = 0; i < 26; i++) {
      if (count1[i] != count2[i]) {
         return false;}}
   return true;}
int main() {
   string str1, str2;
   cout << "Enter the first string: ";
   cin >> str1;
   cout << "Enter the second string: ";
   cin >> str2;
   if (areAnagrams(str1, str2)) {
      cout << "The strings are anagrams." << endl;
   } else {
      cout << "The strings are not anagrams." << endl;}
   return 0;}
```

```
Enter the first string: string
Enter the second string: space
The strings are not anagrams.
```

Question 19)

```
Enter a string: tabish bisht
First non-repeating character: t
Position: 0
```

```
#include <iostream>
#include <string>
using namespace std;
pair<char, int> findFirstNonRepeating(const string& str) {
   int count[256] = {0};
   for (char ch : str) {
      count[ch]++;}
   for (int i = 0; i < str.length(); ++i) {
      if (count[str[i]] == 1) {
         return {str[i], i};}}
   return {'\0', -1};}
int main() {
   string str;
   cout << "Enter a string: ";
   cin >> str;
   auto result = findFirstNonRepeating(str);
   if (result.second != -1) {
      cout << "First non-repeating character: " << result.first << endl;
      cout << "Position: " << result.second << endl;
   } else {
      cout << "No non-repeating character found." << endl;}
   return 0;}
```

Question 20)

```
Enter a string: tabish bisht
First non-repeating character: t
Position: 0
```

```
#include <iostream>
#include <string>
using namespace std;
string longestPalindromicSubstring(const string& s) {
   int n = s.length();
   if (n == 0) return "";
   int start = 0, maxLength = 1;
   auto expandAroundCenter = [&](int left, int right) {
      while (left >= 0 && right < n && s[left] == s[right]) {
         --left;
         ++right;}
      return right - left - 1;};
   for (int i = 0; i < n; ++i) {
      int len1 = expandAroundCenter(i, i);
      int len2 = expandAroundCenter(i, i + 1);
      int len = max(len1, len2);
      if (len > maxLength) {
         maxLength = len;
         start = i - (len - 1) / 2;}}
   return s.substr(start, maxLength);}
int main() {
   string s;
   cout << "Enter a string: ";
   cin >> s;
   string longestPalindrome = longestPalindromicSubstring(s);
   cout << "Longest palindromic substring: " << longestPalindrome << endl;
   return 0;}
```