Cpp Practicals\Q4\Q4.cpp

```cpp
/*
4.  Write a menu driven program to perform string manipulation (without using inbuilt string functions):
    a.  Show address of each character in string
    b.  Concatenate two strings.
    c.  Compare two strings
    d.  Calculate length of the string (use pointers)
    e.  Convert all lowercase characters to uppercase
    f.  Reverse the string
    g.  Insert a string in another string at a user specified position


*/

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

void showAddress(string);  //a
string concatenate(string, string);  //b
void compare(string, string); //c
int stringLength(string); //d
string uppercase(string); //e
string reverse(string); //f
string insertString(string, string, int);//g

int main()
{
    char key;
    while (key != ' ')
    {
        cout<<"String Manipulation Program : Press a-g to manipulate strings, press spacebar to exit"<<endl;
         cout<<"       a.   Show address of each character in string"<<endl;
         cout<<"       b.   Concatenate two strings. "<<endl;
         cout<<"       c.   Compare two strings "<<endl;
         cout<<"       d.   Calculate length of the string (use pointers) "<<endl;
         cout<<"       e.   Convert all lowercase characters to uppercase "<<endl;
         cout<<"       f.   Reverse the string "<<endl;
         cout<<"       g.   Insert a string in another string at a user specified position"<<endl;

        char response;
        cout<<"Enter your response : ";
        cin>> response;

        switch (response)
        {
            case 'a' :
            {
                string str;
                cout<<"Enter a string : ";
                cin>>str;
                showAddress(str);
                break;

            }
```

```
55
56              case 'b':
57              {
58                  string str1, str2;
59                  cout<<"Enter first string : ";
60                  cin.ignore();
61                  getline(cin, str1);
62                  cout<<"Enter second string : ";
63                  getline(cin, str2);
64                  string concinated = concatenate(str1, str2);
65                  cout<<concinated<<endl;
66                  break;
67
68              }
69
70              case 'c':
71              {
72                  string str1, str2;
73                  cout<<"Enter first string : ";
74                  cin.ignore();
75                  getline(cin, str1);
76                  cout<<"Enter second string : ";
77                  getline(cin, str2);
78                  compare(str1, str2);
79                  break;
80
81              }
82
83              case 'd':
84              {
85                  string str;
86                  cout<<"Enter a string : ";
87                  cin>>str;
88                  int len = stringLength(str);
89                  cout << len<<endl;
90                  break;
91              }
92
93              case 'e' :
94              {
95                  string str;
96                  cout<<"Enter a string : ";
97                  cin>>str;
98                  string upper_str = uppercase(str);
99                  cout <<upper_str<<endl;
100                 break;
101             }
102             case 'f':
103             {
104                 string str;
105                 cout<<"Enter a string : ";
106                 cin>>str;
107                 string reversed_str = reverse(str);
108                 cout<<reversed_str<<endl;
109                 break;
110             }
111             case 'g':
112             {
113                 string str1, str2;
114                 int pos;
```

```
                cout<<"Enter first string 1 : ";
                cin.ignore();
                getline(cin, str1);
                cout<<"Enter second string 2 : ";
                getline(cin, str2);
                cout<<"Enter position where you want to insert string 2 : ";
                cin>>pos;
                string newStr = insertString(str1, str2, pos);
                cout<<newStr<<endl;
                break;
            }
            default:
            {
                break;
            }
        }
    }
}

void showAddress(string str)
    {
        for(int i = 0; i < str.length(); i++)
        {
            cout<<"Position of "<<  str[i]<<": "<< (void*) str[i] <<endl;
        }
    }

string concatenate(string str1, string str2)
{
    string conc;
    conc = str1 + str2;
    return conc;
}

void compare(string str1, string str2)
{
    if(str1 > str2)
    {
        cout<<str1<<" > "<<str2<<endl;
    }
    else if (str1 < str2) {
        cout<<str2<<" < "<<str1<<endl;
    }
    else {
        cout<<str1<<" = "<<str2<<endl;
    }
}



int len(string &x)
{
    int count = 0;
    for (int i : x)
    {
        count++;
    }
    return(count);
}

```

```
int stringLength(string str)
{

    char *sptr;
    sptr=&str[0];

    int count=0;

    int i=0;
    while(*sptr!='\0'){
            sptr++;
            count++;

    }
    return count;
}


string uppercase(string str)
{
    string str_upper;
    for(int i = 0; i < str.length(); i++)
    {
        char letter = str[i];
        str_upper += toupper(letter);
    }
    return(str_upper);
}

string reverse(string str)
{
    string reversed_str;
    for(int i = 0; i < str.length(); i++)
    {
        char letter = str[i];
        reversed_str = letter + reversed_str;
    }
    return(reversed_str);
}

string insertString(string str1, string str2, int pos)
    {
    string newStr;
    for (int i = 0 ; i < pos; i ++)
    {
        newStr += str1[i];

    }
    newStr += str2;
    for (int i = pos ; i < str1.length(); i ++)
    {
        newStr += str1[i];
    }
    return(newStr);

    }


/*

```

```
Output :

String Manipulation Program : Press a-g to manipulate strings, press spacebar to exit
       a.       Show address of each character in string
       b.       Concatenate two strings.
       c.       Compare two strings
       d.       Calculate length of the string (use pointers)
       e.       Convert all lowercase characters to uppercase
       f.       Reverse the string
       g.       Insert a string in another string at a user specified position
Enter your response : a
Enter a string : Apple
Position of A: 0x41
Position of p: 0x70
Position of p: 0x70
Position of l: 0x6c
Position of e: 0x65
String Manipulation Program : Press a-g to manipulate strings, press spacebar to exit
       a.       Show address of each character in string
       b.       Concatenate two strings.
       c.       Compare two strings
       d.       Calculate length of the string (use pointers)
       e.       Convert all lowercase characters to uppercase
       f.       Reverse the string
       g.       Insert a string in another string at a user specified position
Enter your response : b
Enter first string : Apple
Enter second string : Mango
AppleMango
String Manipulation Program : Press a-g to manipulate strings, press spacebar to exit
       a.       Show address of each character in string
       b.       Concatenate two strings.
       c.       Compare two strings
       d.       Calculate length of the string (use pointers)
       e.       Convert all lowercase characters to uppercase
       f.       Reverse the string
       g.       Insert a string in another string at a user specified position
Enter your response : c
Enter first string : Pizza
Enter second string : Burger
Pizza > Burger
String Manipulation Program : Press a-g to manipulate strings, press spacebar to exit
       a.       Show address of each character in string
       b.       Concatenate two strings.
       c.       Compare two strings
       d.       Calculate length of the string (use pointers)
       e.       Convert all lowercase characters to uppercase
       f.       Reverse the string
       g.       Insert a string in another string at a user specified position
Enter your response : d
Enter a string : Apple
5
String Manipulation Program : Press a-g to manipulate strings, press spacebar to exit
       a.       Show address of each character in string
       b.       Concatenate two strings.
       c.       Compare two strings
       d.       Calculate length of the string (use pointers)
       e.       Convert all lowercase characters to uppercase
       f.       Reverse the string
       g.       Insert a string in another string at a user specified position
```

```
Enter your response : e
Enter a string : Hello
HELLO
String Manipulation Program : Press a-g to manipulate strings, press spacebar to exit
       a.       Show address of each character in string
       b.       Concatenate two strings.
       c.       Compare two strings
       d.       Calculate length of the string (use pointers)
       e.       Convert all lowercase characters to uppercase
       f.       Reverse the string
       g.       Insert a string in another string at a user specified position
Enter your response : f
Enter a string : Avinash
hsanivA
String Manipulation Program : Press a-g to manipulate strings, press spacebar to exit
       a.       Show address of each character in string
       b.       Concatenate two strings.
       c.       Compare two strings
       d.       Calculate length of the string (use pointers)
       e.       Convert all lowercase characters to uppercase
       f.       Reverse the string
       g.       Insert a string in another string at a user specified position
Enter your response : g
Enter first string 1 : Avinash
Enter second string 2 : Shrivastava
Enter position where you want to insert string 2 : 7
AvinashShrivastava


*/
```