**CS-1004 Object Oriented Programming Fall-2021**

**ASSIGNMENT-01**

**Section (All)**

**Submission Deadline: 01 oct, 2021 11:59 pm**

Instructions:

1. Assignments are to be done individually. You must complete this assignment by yourself. You cannot work with anyone else in the class or with someone outside of the class. The code you write must be your own and you must understand each part of your code. You are encouraged to get help from the instructional staff through google classroom.
2. Do not use any String or math libraries (such as cmath, cstring, string etc) and also do not use built-in function (such as pow etc). **Caution**: zero marks will be awarded.
3. Do not edit **Function Prototypes**. **Caution**: zero marks will be awarded.
4. The usage of string is strictly prohibited.
5. Your code must be **generic**.
6. Marks distribution and test Cases are provided for each question. Your code will be evaluated with **similar test cases**. If the required output is generated, you will be awarded full marks. Failing to generate the correct output will result in zero marks. Total Marks: 150.
7. For **PrintPattern** questions, the output should be properly displayed and well presented. There will be no gtests for **PrintPattern** questions.
8. **Plagiarism**: Plagiarism of any kind (copying from others, copying from the internet, etc) is not allowed. If found plagiarized, you will be awarded zero marks in the assignment. Repeating such an act can lead to strict disciplinary actions and failure in the course.
9. Please start early otherwise you will struggle with the assignment.
10. **Test cases:** Test cases (in gtest) will be shared with you on Google Classroom. **We will be running your code against our test cases, and a test case failure or a segmentation fault/incorrect result will result in zero marks.**
11. **Submission Guidelines**: Dear students, we will be using **auto-grading tools (gtest)**, so failure to submit according to the below format would result in zero marks in the relevant evaluation instrument.
    1. Make your own file named submission.cpp. Please don’t include the main function while submitting the file. And don’t remove **test cases** (in testcases.cpp) or **function prototypes** (in submission.cpp).
    2. Your submission.cpp file must contain your name, student-id, and assignment # on the top of the file in the comments.
    3. Move you submission.cpp in one folder. The folder must contain only submission.cpp file (no binaries, no exe files etc.,). If we unable to download your submission due to any reason you will be awarded zero mark.
    4. Run and test your program on a lab machine before submission. If there is a syntax error, zero marks will be awarded in that specific question.
    5. Rename the folder as ROLL-NUM\_SECTION (e.g. 20i-0001\_A) and compress the folder as a zip file. (e.g. 20i-0001\_A.zip). Only zip file will be acceptable.
    6. Submit the .zip file on Google Classroom within the deadline.
    7. Submission other than Google classroom (e.g. email etc.) will not be accepted.
    8. The student is solely responsible to check the final zip files for issues like corrupt files, viruses in the file, mistakenly exe sent. If we cannot download the file from Google classroom due to any reason it will lead to zero marks in the assignment.

Note: Follow the given instruction to the letter, failing to do so will result in a zero.

Q1: String manipulation functions

Write the implementation of the following functions:

Marks:10

int Strlen(char\* s1)

/\*Returns the length of the string in number of characters. \*/

{

}

Example:

s1 => "This is the first OOP assignment"

return value will be 32

Marks:10

char\* Strcpy(char\* s1, const char \*s2)

/\*Copies string s2 into array s1. The value of s1 is returned. \*/

{

}

Example:

s1 => "This is OOP assignment"

s2 => "01"

return char\* => “This is OOP assignment01”

Marks:10

int StrCmp(const char\* s1, const char \*s2)

/\*Compares the string s1 with the string s2.

The function returns 0 , -1 or 1 if s1 is

equal to , less than or greater

than s2 , respectively. \*/

{

}

Example01:

s1 => "Happy New Year"

s2 => "Happy New Year"

StrCmp(s1, s2) will return 0

Example02:

s1 => "Happy New Year"

s2 => "Happy Holidays"

StrCmp(s1, s2) will return -1

Example03:

s1 => "Happy Holidays"

s2 => "Happy New Year"

StrCmp(s1, s2) will return 1

Marks:30

void StrTok(char\* s1, const char s2, char\*\*& listOfTokens , int& size)

/\*A call to StrTok breaks string s1 into

"tokens" ( logical pieces such as words

in a line of text) separated by character

contained in const char s2. size parameter

will be updated number of tokens and save

actual tokens in char\*\*& listOfTokens.

\*/

{

}

Example:

s1 => "This is the first OOP assignment"

s2 => ‘ ’ (it’s a space character)

output

listOfTokens => {“This”, “is”, “the”, “first”, “OOP”, “assignment”}

size => 6

Marks:30

bool StrFind(char\* s1, char \*s2,int\*& listOfOccurrences, int& size)

/\*Searches the string s1 for the all the occurrences

of the string s2. Save starting indexes in array (listOfOccurrences). And save updated size of array listOfOccurrences also return true. But if s2 not found returns false. \*/

{

}

Example:

s1 => "substrstringstr"

s2 => “str”

output

bool => true

listOfOccurrences => {3, 6, 12}

size => 3

bool StrSwapIn2DArray (char\*\* s1, int numberOfRows, int swapIndex0, int swapIndex1)

/\*swap the string of swapIndex0 and

swapIndex1 within 2d char array s1

return true if indexes are correct

and swapping is successful and return

false if swapping is not successful

(indexes are incorrect) \*/

{

}

Example:

s1 => {“This”, “is”, “the”, “first”, “OOP”, “assignment”}

numberOfRows => 6

swapIndex0 => 2

swapIndex0 => 5

output

bool => true

s1 => {“This”, “is”, “assignment”, “first”, “OOP”, “the”}

Q2: Recursive Functions

Marks:10

1. Write a recursive function find that finds given target value in the array. If value is not found your function must return -1, otherwise it should return the index of array where the value was found. Your function prototype must be as follows:

**int find(int array[], int length, int target);**

Marks:10

1. Write a recursive function replace that changes all the occurrences of character 'f' in String 's' to character 't' and then returns the changed string. For example,

replace (steve, ‘e’, ‘a’) // it will return stava

replace (radar,’a’,’o’) // it will return rodor

**Prototype**:

**void replace(char\* s1, char ch1, char ch2);**

Marks:10

1. Write a recursive function PrintPattern1 to print pattern that takes two integer arguments **n** and **k**. **n** is the starting number while **k** is the ending limit.

Example:

PrintPattern1(1, 5,’@’,’+’) would print the following pattern.

1@2@@3@@@4@@@@5@@@@@4++++3+++2++1+

**Prototype**:

**void PrintPattern1(int n, int k, char ch1, char ch2);**

Marks:10

1. Write a recursive function PrintPattern2 that receives two arguments: (i) a character ch; (ii) number of lines and print the pattern.

Example:

PrintPattern2('\*',5) will print the following pattern

**\***

**\*\***

**\*\*\***

**\*\*\*\***

**\*\*\*\*\***

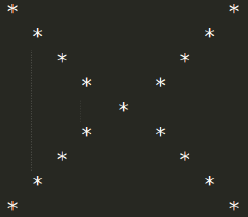
**Prototype**:

**void PrintPattern2(char ch, int n);**

Marks:20

1. Write a C++ recursive function PrintPattern3 to print following pattern using recursion. No loops allowed whatsoever, and you can write maximum two functions apart from main function. For example, calling your function with these argument PrintPattern1(1,5) should print following pattern. Your function prototype must be as follows recursive function.

**void PrintPattern3(int start, int end);**



**Happy Coding 😊**