

CMPT125, Fall 2021

Lab exam - Version 2

Wednesday, November 17, 2021, 10:30am-11:20am

You need to implement the functions in ***labexam.c***.

Submit only the **.c** file to Canvas

Canvas Assignment - Lab Exam 10:30-11:20.

You have 50 minutes to solve all 3 problems.

The maximal score is 20 points.

The exam will be graded both **automatically** and by **reading your code**.

You can run your code using

```
>> make
```

```
>> ./run_test_v2
```

Correctness: Make sure that your code compiles without warnings/errors, and works as expected.

Readability: Your code should be readable. Add comments wherever necessary. If needed, write helper functions to break the code into small, readable chunks.

Compilation: Your code **MUST** compile in CSIL with the Makefile provided. If the code does not compile in CSIL, the grade on the assignment is 0 (zero). Even if you can't solve a problem, make sure it compiles.

Helper functions: If necessary, you may add helper functions to the .c file.

main() function: do not add main(). Adding main() will cause compilation errors, as the main() function is already in the test file.

Using printf()/scanf(): Your function should have no unnecessary printf() statements. They may interfere with the automatic graders.

Warnings: Warnings during compilation will reduce points. More importantly, they indicate that something is probably wrong with the code.

Testing: An example of a test file is included.

Your code will be tested using the provided tests as well as additional tests.

You are *strongly encouraged to write more tests* to check your solution is correct, but you don't need to submit them.

Question 1 [6 points]

Write a function that gets a 2-d array of ints `arr[height][width]`, and returns an array `MAX` of length `height` such that `MAX[i]` contains the maximal value in the i 'th row of the array. For example

- On input $\begin{Bmatrix} 1, & 2, & 3, \\ 4, & 7, & -2, \\ 9, & 10, & 12, \\ 1, & 5, & 4 \end{Bmatrix}$;
the function returns $\{ 3, 7, 12, 5 \}$

```
// gets a 2d array of ints
// returns array MAX with MAX[i] = the maximal value in i'th row
int* max_row(int height, int width, const int ar[height][width]);
```

Question 2 [7 points]

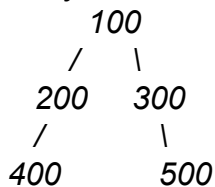
Write a function that gets a string, and a char `c`, and returns the number of substrings consisting only of `c`. In the examples below the subsequences are marked bold.

- `count_substrings("HELLO WORLD", 'L')` needs to return 2
- `count_substrings("AABBCD A* * AAAACabcd ", 'A')` needs to return 3.
- `count_substrings("What's my line", '*')` needs to return 0

```
// counts subsequences consisting only of c's
int count_substrings(const char* str, char c);
```

Question 3 [7 points]

Write a function that gets a root of a binary tree, and returns the number of nodes with exactly one child.



The function returns 2 because there are two vertices with exactly one child: 200, 300.

```
// gets a root of a binary tree
// returns the number of nodes with exactly one child
int count_nodes_one_child(const BTreeNode* root);
```