

## CMPT125, Fall 2022

Lab exam - 10:30am-11:20am

Thursday, November 10, 2022

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

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

Coursys Assignment - **Lab Exam D101-D102 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
```

**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 not have any 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 an array of strings, and reverses the array. For example:  
Suppose we have an array `ar = ["Hello", "Hello", "hi", "Hi", "ABC"]`.  
Then, after applying `rev_arr(ar, 5)`, `ar` becomes `["ABC", "Hi", "hi", "Hello", "Hello"]`.

```
// gets an array of strings of length n, and reverses the array
void rev_arr(const char** ar, int n);
```

### Question 2 [7 points]

Write a function that gets a sorted array of ints of length  $n > 0$ . It moves all zeros to the end of the array, keeping the relative order of all other numbers unchanged. The function returns the number of zeros.

- Suppose we have an array `ar = [1,2,0,3,0,3,1,0]`.
- When applying `move_zeros(ar, 8)`, `ar` becomes `[1,2,3,3,1,0,0,0]`, and returns 3.

```
// the function gets ar of length n>0, and moves all zeros
// to the end of the array, and returns the number of zeros.
int move_zeros(int* ar, int n);
```

### Question 3 [7 points]

Write a function that gets a linked list of ints and returns a copy of the list.  
See the file `lib/LL.h` for the functions you can use,

```
// the function gets a linked list of ints
// and returns a new copy of the list
// assumption: list != NULL
LL_t* LL_copy(LL_t* list);
```