

CMPT125 D107-108, Fall 2025
Lab exam - Thursday, 2:30pm - 3:20pm
November 20, 2025
You need to implement the functions in ***labexam.c***.
Submit only the **.c** file to Coursys
Coursys Assignment - **Lab Exam D107-D108**.

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 two parameters $start \leq end$ such that $0 \leq start \leq end < \text{length of ar}$.

The function reverses the subarray $ar[start...end]$.

For example

- On input $ar = ["Hello", "****", "", "AB2C", "Hi!7", "123"]$, $start = 2$, $end = 4$, when the function returns the array will be $["Hello", "****", "Hi!7", "AB2C", "", "123"]$.

```
// The function gets an array ar of strings and two parameters start <= end
// and reverses the subarray ar[start...end]
// Assumption: 0 <= start <= end < strlen(str)
void reverse_subarray(char** ar, int start, int end);
```

Question 2 [7 points]

Write a function that gets a string str and a string pattern, and finds the last appearance of pattern in str .

The function returns the index in str of the beginning of the last appearance

If pattern is not in str , returns -1. For example,

- On input $str = "abcdabXBabc-bc"$ and $pattern = "abc"$, the function returns 8.

```
// The function gets two strings, str and pattern.
// It finds the last appearance of pattern in str
// and returns the index in str of the beginning of the last appearance
// If pattern is not in str, the function returns -1
int str_find_last(const char* str, const char* pattern);
```

Question 3 [7 points]

Write a function that gets a linked list of ints and a predicate, and returns the number of items in the list satisfying $pred(item) == true$.

For example,:

- On input $3 \rightarrow 2 \rightarrow 7 \rightarrow 8 \rightarrow 10 \rightarrow 5$ and $pred = is_even$ the function returns 3

See the file `lib/LL.h` for the functions you can use.

```
// the function gets a Linked List of ints and a predicate
// and returns the number of nodes satisfying pred(node->data)==true
// when the function returns the Linked List needs to be in its initial state
int LL_countif(LL_t* lst, bool(*pred)(int));
```