

CMPT125, Spring 2022  
Lab exam - D201-D202

Tuesday, March 15, 2022, 9:30am-10:20am  
You need to implement the functions in **labexam.c**.  
Submit only the **.c** file to Coursys  
Coursys Assignment - Lab Exam 9:30-10:20.

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

**Submit only labexam.c:** Please make sure to submit the file to the *correct section* in Coursys.

**Correctness:** Your file must compile without warnings/errors, and work 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 *labexam.c* file.

**main() function:** do not add main() to labexam.c. 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 an array of strings of length  $n$ . It changes all lowercase vowels in all strings to an asterisk, and returns the total number of modified chars. For example:

- `vowels2asterisks(["HeLLo", "hi", "yes", "NO"], n=4)` should modify the strings to `["H*LL*", "h*", "y*s", "NO"]` and return 4.

```
// the function gets an array of strings of length n
// and modifies each string by changing all *lower case* vowels into asterisk
// and returns the total number of modified chars
// The vowels are: {a,e,i,o,u}
int vowels2asterisks(char* ar[], int n);
```

### Question 2 [7 points]

Write a function that gets an array of ints of length  $n > 0$ , and returns the longest consecutive sequence of equal numbers. For example:

- `longest_equal_seq( [1,8,3,3,8,1 8] )` returns 2 (two consecutive 3's).
- `longest_equal_seq( [1,2,4,4,4,7,1,1,8,1,1,100] )` returns 3 (three consecutive 4's).
- `longest_equal_seq( [1,2,3,4,5,99] )` returns 1 (equal sequences are of length 1).
- `longest_equal_seq( [1,1,2,2,2,2,2,1,1,30,1,1,1,40,1,1] )` returns 5.
- 

```
// the function gets an array of ints of length n>0
// and returns the longest consecutive sequence of equal numbers
// for example, on input [1,2,4,4,4,7,1,1,8,1,9,1,1]
// the function returns 3 because the number 4 appears three times in a row
int longest_equal_seq(const int* ar, int n);
```

### Question 3 [7 points]

Write a function that gets pointers to two queues of ints and swaps their content. If both pointers point to the same queue, the function doesn't do anything. See `lib/queue.c` and `lib/queue.h` for details.

```
// gets two queues, and swaps their content
void queues_swap(queue_t* q1, queue_t* q2);
```