



## CSE 1322L Final Spring 2021

- You will use repl.it to write, test, and debug your code during this test.
- **YOU MUST PASTE (Ctrl v) THE COMPLETE PROGRAM CODE INTO THE D2L ANSWER BOX**
  - Paste all classes, methods, attributes and includes/uses statements into the D2L answer box below each question. We only grade what you put into D2L, nothing in repl.it will be graded.
  - Do not paste the URL of the repl.it.
- All answers must be your own, without the assistance of others.
- While taking this test you may not use any resources. This includes but is not limited to:
  - You cannot converse or listen to anyone while taking the test.
  - You cannot use electronics of any kind, including cell phones, tablets, other computers, earbuds, headphones, calculators or watches.
  - No paper of any kind, including scratch paper, notes, or book.
  - You cannot use any resources from the internet other than the repl.it we provide
- You must remain in view of the camera with adequate lighting for the duration of the testing period. The camera should be centered on your face so that the monitoring software can detect your face during the test.
- Anyone who violates these rules will receive a 0 on this test.
- You will only have one (1) attempt to take the test, make sure you can commit to the full amount of time as there is no stopping the timer for any reason.
- You will have 90 minutes to take this test. Use your time wisely, answer the easiest questions first, come back to the more difficult ones.
- **If you are in a Java section (WJ1, J01-5) you must write in Java. If you are in a C# section (W#1, #01-04) you must write in C#. What you turn in must compile.**
- Partial credit will be awarded where appropriate.

Name: \_\_\_\_\_

Date: \_\_\_\_\_

1) (15 points)

Create a class called Note, which has two private attributes: theNote (String) and priority (int).

Write a constructor which takes the two parameters and sets them.

Create methods getNote() and getPriority() which return the note and priority respectively.

Overwrite toString() (Java) / ToString() (C#) so it returns a string with the priority followed by a comma, followed by theNote text.

2) (10 points)

Create an interface INotable which has two methods.

The first method is next\_todo() which takes no parameters and **returns** a Note object.

The second method is save() which takes a single String parameter filename and **returns** a boolean indicating success or failure.

3) Next create a Notebook class in parts:

a. (5 points) Create a class called Notebook that implements INotable.

b. (5 points) Create an ArrayList or List that can hold Notes objects in the Notebook class.

c. (15 points) Create a method addNote() which takes a string and priority and adds a Note object to the ArrayList or List.

d. (20 points) Implement the save() method in your Notebook class. The method must return a boolean indicating success. It will take in a single String parameter called filename. It should open the filename file for writing, and write all of the notes into the file, one per line. You'll likely want to use the toString() (Java) / ToString() (C#) method in the Note class to format the line.

The method **must handle any exceptions** that come up while attempting to write the file. It must return true as long as no exceptions occurred.

e. (15 points) Implement next\_todo() in your Notebook class. The method returns a *Note object* and takes no parameters. The method **should return the Note with the highest priority** (i.e. highest value in the priority attribute).

4) (15 points) Implement a class called Monster which has 2 attributes, x\_location and y\_location which are both integers. They should both be initialized to 0. The Monster class should have a run() method that adds 100 to the x\_location attribute and 200 to the y\_location attribute.

Write a main method where you create two threads of the class `Monster`. Start both threads.