

**15-112 Spring 2018 Quiz 5**

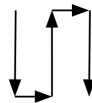
Up to 20 minutes. No calculators, notes, books, or computers. Do not use recursion. Show your work!

1. (25 points) **Code Tracing:** Indicate what the following program prints. Place your answers (and nothing else) in the box to the right of the code.

```
import copy
def ct5(a):
    b = a
    c = copy.copy(a)
    d = copy.deepcopy(a)
    a = a + [ 10 ]
    b[1][0] = "one"
    c[0] = "two"
    d[0].extend(d[1])
    d[1][1] = 12
    print("a:", a)
    print("b:", b)
    print("c:", c)
    print("d:", d)
e = [ [ 2, 4 ], [ 6, 8 ] ]
ct5(e)
print("e:", e)
```

2. (25 points) **Free Response:** Write the function `makeTwistyString` which takes a 2D list of characters as input and returns the characters combined into a single string by following a specified path. The path should traverse the columns of the list by going down the first column, then up the second, then down the third, etc., until it reaches the last column. You are guaranteed that the provided list will be rectangular (not ragged). An example is shown below:

```
lst = [ [ "A", "F", "G" ],
        [ "B", "E", "H" ],
        [ "C", "D", "I" ] ]
```



```
result = "ABCDEFGHI"
```

3. (15 points) **Short Answer:** Assume that you want to build an animation that lets the user play the game Tic-Tac-Toe. Write up **one** element of a Model, View, and Controller for this game. No code is needed, just describe the element in your own words.

4. (35 points) **Free Response:** Using our animation framework and assuming `run()` is already written, write the `init(data)`, `mousePressed(event, data)`, `keyPressed(event, data)`, and `redrawAll(canvas, data)` functions for an animation which has the following elements:
1. A yellow circle with a radius of 50 pixels that starts in the middle of the screen
  2. Red text saying "Hello!" in the bottom-left corner of the screen, with all of the text onscreen.
  3. When the user clicks the mouse, the circle immediately moves to be centered on the clicked location
  4. Every time the user types a key, the typed key is added to the end of the text in the bottom-left corner. Text should never disappear off the left or bottom sides of the screen, but it can disappear off the right side if the user adds enough characters. Don't worry about handling non-visible characters for now.

**Note:** to save time, feel free to abbreviate `canvas`, `data`, and `event` as `c`, `d`, and `e`.

**YOU MAY CONTINUE WRITING ANIMATION CODE ON THIS PAGE.**

USE THIS PAGE FOR SCRAP WORK. WORK ON THIS PAGE WILL NOT BE GRADED.