

Quiz 7 Free Response (60 pts)

Time Limit: 25 mins

Name: _____ AndrewID: _____

You can run code when completing this part of Quiz 7.

Time limit: 25 mins

Free Response 1: busiestStudents (25 pts)

Background: We will represent course rosters using a dictionary mapping course numbers to sets of students in the course. For example:

With that in mind, write the function `busiestStudents(rosters)` that takes a non-empty dictionary of rosters such as the one above, and returns a set of the students who are taking the most courses. In the example above, Claire and Elena are both taking 3 courses, which is the most, so `busiestStudents(rosters)` returns the set `{'Claire', 'Elena'}`.

Note: You may only use material that was covered in units 1-7. Thus, do not use OOP or recursion.

Test cases:

```
def testBusiestStudents():
    rosters = {
        '15-112': {'Amy', 'Bob', 'Claire', 'Dan'},
        '18-100': {'Amy', 'Claire', 'Elena', 'Fred'},
        '21-127': {'Claire', 'Elena', 'George'},
        '76-101': {'Bob', 'Elena', 'Fred'},
    }
    assert( busiestStudents(rosters) == {'Claire', 'Elena'} )

    rosters = {
        'A': {'x', 'y'},
        'B': {'x', 'z'},
        'C': {'x', 'y'},
    }
    assert( busiestStudents(rosters) == {'x'} )

    rosters = {
        'M1': {'a', 'b', 'c'},
        'M2': {'a'},
        'M3': {'b'},
    }
    assert( busiestStudents(rosters) == {'a', 'b'} )
```

Write your answer below.

Free Response 2: Dragging Grids (35 pts)

Write an animation such that:

- The app starts with a blank canvas (except for the title, which is in the starter code).
- A 5x5 grid is created when the mouse is pressed and dragged.
 - When the mouse is pressed, that location becomes one corner of a 5x5 grid.
 - As the mouse is dragged, the current mouse location is the opposite corner of the grid.
 - If the mouse is released, the grid stays in the last position it was dragged to. Once a new mouse press occurs, a new grid is created. The old grid does not remain on the screen.

Notes:

- `drawBoard` has been included for you in the starter code. You should call this function, but you do not need to edit it.
- You must implement the functions `getBoardWidthHeight`, `getBoardLeftTop`, `getCellWidthHeight`, and `getCellLeftTop` for the `drawBoard` function to work.

Hint: If you can't figure out how to solve the general problem, start by assuming that the mouse press location is always the top-left of the grid, and the mouse drag location is always the bottom-right. You will get at least 25 points for correctly solving this version of the problem.

You may use any topics covered in class so far. You may not use OOP or recursion.

Write your answer below.