Name:_____ Section:___ Andrew Id: _____

15-112 Fall 2014 Quiz 7

*** 20 minutes. No calculators, no notes, no books, no computers.**
*** You may not discuss any portion of this quiz with anyone until after 5pm today.**
*** SHOW YOUR WORK, CIRCLE YOUR ANSWERS.**

1. **Short Answers: Classes** [35 pts; 5 pts each]
The following questions concern this code:

```
class A(object):
    def __init__(self, b, c):
        assert((type(b) == int) and (type(c) == str))
        (self.b, self.c) = (b, c)
```

Hint: One of these questions requires a static method.

1) Write one method in A so (x == y) works as expected when at least one of x or y is an instance of A.

2) Write one method in A so repr(x) works as expected when x is an instance of A.

3) Write one line of code that uses the standard, general test to verify that repr() works as expected with instances of A.

4) What would we expect the return type to be for a call to a.__hash__() if a is an instance of A?

Reminder: These questions concern this code:

```
class A(object):
    def __init__(self, b, c):
        assert((type(b) == int) and (type(c) == str))
        (self.b, self.c) = (b, c)
```

Hint: One of these questions requires a static method.

5) Write one method to make this True: ((A(1,'b') + 2) == A(3, 'b')) and ((A(1,'b') + 'c') == A(1,'bc'))

6) Write one method to make this True: A.default() == A(42,'a')

7) Write code (not just one method) to create a variable z that makes this True:
(isinstance(z, A) and (type(z) != A))

2. **Free Response: TetrisR** [35 pts]
Write a **subclass** of our OOP-based Tetris, called TetrisR, which is identical to Tetris, except when the user presses the 'r' key, the current piece moves (without rotating) as far right as possible without colliding with an occupied cell or moving off the board. Hint: you may want to use super(TetrisR, self) somewhere in your solution.

3. **Free Response: SnakeF** [30 pts]
   Write a **subclass** of our OOP-based Snake, called SnakeF, which is identical to Snake, except when the user clicks the mouse anywhere in the left half of the canvas, the food is moved to a random new location (but not on the snake, and not where the food was when they clicked the mouse). You may ignore the case when there is no such location. Also, you should ignore mouse clicks in the right half of the canvas.

Bonus/Optional: [3 pts] Indicate what this will print:

```
class B(object):
    def __init__(self, z, x=42):
        if (type(z)==B): self.x = z.x * 10
        elif (isinstance(z,B)): self.x = z.x + 1
        else: self.x = -x
    def __str__(self): return str(self.x)
class B2(B):
    def __str__(self): return "<%s>" % super(B2,self).__str__()
class B3(B2):
    def __init__(self, z, x=42): self.x = x
print B(B(5),6), B(B2(5),6), B3(B(5),6)
```