Name:

Instructions

- For maximum credit you must write your name on *all pages* of the exam.
- You have the full class period to complete the exam. Extra time will only be given for students with learning disablilities officially recognized by the University.
- Laptops, netbooks, smartphones, video game consoles, etc. may not be used during the exam. Turn them off/put them to sleep and keep them out of view. Simple calculators are allowed, but should mostly be unecessary. Whether or not a device qualifies as a simple calculator is entirely up to the instructor.
- No communicating with any other student or looking at any other student's test while taking the exam.
- You are free to leave once you turn in your exam.

Multiple Choice

Circle the letter of the most correct answer among the choices. (2 points each.)

1. Choose the invalid line of code:

   (a) `99 + 8`
   (b) `76trombones = '76 trombones'`
   (c) `stuff = 'things'`
   (d) `return num_dogs * food_quota`
   (e) All of the above lines are valid

2. Choose the invalid line of code:

   (a) `choice_specs = 50`
   (b) `50 == 25 * 2`
   (c) `doc + scratch`
   (d) `choice-specs = 50`
   (e) All of the above lines are valid

3. Choose the invalid line of code:

   (a) `counter = * 8 7`
   (b) `robots == fish`
   (c) `balloons = 99`
   (d) `balloons == 99`
   (e) All of the above lines are valid

Name:

4. Choose the invalid line of code:

   (a) `stuff = $`

   (b) `color = 'blue'`

   (c) `frog == glenn`

   (d) `hungry_like(the_wolf)`

   (e) All of the above lines are valid

5. Choose the invalid line of code:

   (a) `tmp = tmp/10000`

   (b) `math.sqrt(2)`

   (c) `vriska = 'dead'`

   (d) `9 * 11`

   (e) All of the above lines are valid

6. Choose the invalid line of code:

   (a) `vormav == bad + man`

   (b) `vormav = bad + man`

   (c) `99 == 11 + 88`

   (d) `99 = 11 + 88`

   (e) All of the above lines are valid

7. Choose the invalid line of code:

   (a) `house == bitter + man`

   (b) `tmp = tmp * 5`

   (c) `type(i)`

   (d) `return type(i)`

   (e) All of the above lines are valid

8. Choose the invalid line of code:

   (a) `a**2 + b**2 == c**2`

   (b) `circus = clown car + unicycle`

   (c) `return monkeys`

   (d) `return monkeys + peanuts`

   (e) All of the above lines are valid

Name:

9. Choose the invalid line of code:

   (a) `10,000 / 100`

   (b) `return 10 + 1`

   (c) `eighty_eight = 66 + 22`

   (d) `eighty_eight == 66 + 22`

   (e) All of the above lines are valid

10. Choose the invalid line of code:

    (a) `answer == a % 5`

    (b) `answer = a % 100`

    (c) `answer = a % 6.0`

    (d) `answer = a % 33`

    (e) All of the above lines are valid

Short Answer

Write a short (one word will suffice for these questions) answer for each question. (2 points each.)

11. What is the type of `5`?

12. What is the type of `'Squiddles'`?

13. What is the type of `None`?

Name:

Use the following peice of code for questions 14 **thru** 20

```
def f(num):
    apples = 99
    oranges = 56.5
    comp = apples < oranges
    some_stuff = 'Some stuff'
    a_value = oranges * 5
    another_value = apples * 5
    more_stuff = some_stuff * 5
    yet_another_value = apples * 5.0
```

14. What is the type of `apples`?

15. What is the type of `a_value`?

16. What is the type of `yet_another_value`?

17. What is the type of `more_stuff`?

18. What is the type of `comp`?

19. What is the type of `f`?

20. What is the type of `type(apples)`?

Name:

State

21. what is the state diagram after the execution of the following block of code? (10 points)

```
thing = 64
number = 128
j = 3
large_num = 13234 * 33288 + 76533/233
junk = 'Bluh'
large_num = 2**10
more_junk = junk * j
larger_num = large_num + number + thing
number = '99'
thing = ' Red Balloons'
j = larger_num - 192
thing = number + thing
```

Name:

22. what is the stack diagram for the execution of the following block of code? (10 points)

```
def f(x):
    return x ** 2

def g(x):
    y = x + x
    return y + x

def h(x):
    return x / 2

answer = g(f(h(10)))
```

Name:

Code Trace

Using the following code, determine whether or not the assert statements in 23 **and** 24 **hold. If the assertion fails, write down what the actual value of the function is and then write out a fixed version of the function such that the assertion does hold. (10 points each.)**

```
def phrase1():
    return 'Obi-Wan Kenobi killed'

def phrase2():
    return 'I AM'

def phrase3():
    return "smellin' like the rose"

def guy1():
    return 'your father'

def guy2():
    return 'Darth Maul'

def plot_twist():
    return 'No, ' + phrase1() + ' ' + guy1()

def line1():
    return phrase2() + ' ' + phrase3()

def stp():
    line2 = 'that somebody gave me'
    line3 = 'on my birthday deathbead!'

    return line1() + '\n' + line2 + '\n' + line3

def a_statement():
    return phrase1() + ' ' + guy2()
```

Name:

23. `assertEqual(plot_twist(), 'No, I AM your father')`

24. `assertEqual(line1(), "I AM smellin' like the rose")`

Name:

Code Writing (20 points)

Write a function `factorial_difference` that takes an integer n and returns $n! - (n-1)! - (n-2)! - ... - 2! - 1!$. **Hint:** It may be useful to first define a `factorial` function which computes $n!$ for some n and then use that in `factorial_difference`