1. (6) Check the appropriate choice noting which of the following are valid identifiers and which are not.

Valid  Invalid

This_is_a_good_identifier
This&That
7UP
K2O2
StrangeOne
Good4You!

2. (4) What are the values represented by the following arithmetic expressions. Make sure that the value is in the proper form to indicate its type (int or float).

a) \(3 + 6 \times 4 - 7 \div 3\) \(= 26\)  
b) \((3 + 4) \div 2 \times 5\) \(= 15\)  
c) \(\text{math.sqrt}(9) \times 4 + 3 \div 4.0\) \(= 12.75\)  
d) \((14 \times 5) \div 5.0\) \(= 0.8\)

3. (4) Rewrite the following mathematical expressions into an equivalent Python expression. You may assume that the math library has been imported. Do NOT attempt to transform the expression to an equivalent expression. Rewrite them exactly as they have been presented.

a) \(x^3 - 5ab\)

\[x^3 - (5 \times a \times b)\]

b) \(4\pi + 6 \div 4\) (make sure you get a floating point result)

\[\frac{4 \times \text{math.pi} + 6}{4.0}\]

4. (6) Show the list of numbers that would be generated by each of the following range expressions.

a) \(\text{range}(3,10)\)

\([3, 4, 5, 6, 7, 8, 9]\)

b) \(\text{range}(4, 14, 3)\)

\([4, 7, 10, 13]\)

c) \(\text{range}(15, 5, -2)\)

\([15, 13, 11, 9, 7]\)
5. (9) Show the output that would be generated by each of the following program segments.

a)  
   ```python
   x = 3
   y = 12
   for j in range(0, y, x):
       print j,
       print j * j
   print 'done'
   ```

   0 0
   3 9
   6 36
   9 81
   done

b)  
   ```python
   for i in [1, 3, 5, 7]:
       print i, ':', i ** 3
   ```

   1:1
   3:27
   5:125
   7:343

6. (6) What are the values of each of the following assuming that s = "Hello, I have come." Assume `string` has been imported. Each line should use the original value of s.

   ```
   s.capitalize()  Hello, i have come.
   string.capwords(s)  Hello, I Have Come.
   s.replace('T', 'you')  Hello, you have come.
   s.count('e')  3
   s.find(',')  5
   s.lower()  hello, i have come.
   ```

7. (4) Show the output that would be generated by each of the following program fragments.

a)  
   ```python
   msg = ""
   for s in string.split("secret","e"):
       msg = msg + s
   print msg
   ```

   ```
   secret
   ```

b)  
   ```python
   for w in string.split('Now is the winter of our discontent...'):
       print w
   ```

   ```
   Now
   is
   the
   winter
   of
   our
   discontent...
   ```
8. (5) What is output from the following print statements?

a) ".3f %0.3f" % (6.4, 6.45896)  
   \[6.400 \, 6.459\]

b) "%.7f %.7f" % (2.3, 2.34888)  
   \[2.30600 \, 2.34880\]

c) "There are %d %s %d %s" % (3, 'spams', 4, 'you')  
   \[There \, are \, 3 \, spams \, 4 \, you\]

9. (12) What is output from the following loop structures? Use the lines at the right for your answer. There may be extra lines, but the number of lines is sufficient for a correct answer.

a) \[\begin{array}{l}
   x = 1 \\
   \text{while } x \leq 7: \\
   \quad y = 2 \\
   \quad \text{print } x + y, \\
   \quad \text{while } y < 15: \\
   \quad \quad y = (y + x) \times 2 \\
   \quad \quad \text{print } y, \\
   \quad \quad y += 2 \\
   \quad \text{print} \\
   \quad x += 2
\end{array}\]

\[\begin{array}{cc}
   3 & 6 \, 18 \\
   5 & 10 \, 30 \\
   7 & 14 \\
   9 & 18
\end{array}\]

b) \[\begin{array}{l}
   \text{for } k \text{ in range}(2,6): \\
   \quad m = k \times 2 \\
   \quad \text{while } m < 12: \\
   \quad \quad \text{print } "k = \, \, k, \, m = \, m" \\
   \quad \quad m += 4
\end{array}\]

\[\begin{array}{cccc}
   K = 2 & m = 4 \\
   K = 2 & m = 8 \\
   K = 3 & m = 6 \\
   K = 3 & m = 10 \\
   K = 4 & m = 8 \\
   K = 5 & m = 10
\end{array}\]

10. (4) What are the values of the following boolean expressions assuming \(x = 6\), \(y = 9\), \(q = \text{True}\) and \(p = \text{False}\)? Circle the appropriate answer.

a) T  \quad F  \quad (x > y - 4) \text{ and } q \text{ and not } p

b) T  \quad F  \quad \neg(p \text{ or } q) \text{ or } x > y

c) T  \quad F  \quad x > y + 4 \text{ and } q

d) T  \quad F  \quad x < 10 \text{ and } y < 10 \text{ and } q \text{ or } p
11. (5) What is printed by the following code fragment?

```python
myList = []
for i in range (0, 6, 2):
    for k in range(3):
        myList.append(i+k)
print i
print k
print myList
```

12. (6) Write a function that returns the area of a triangle given the length of its three sides as parameters (a, b and c). Use the following formulas for your computation. Make sure the value returned is a float. Assume math has been properly imported.

\[
s = \frac{a + b + c}{2}
\]

\[
area = \sqrt{s(s - a)(s - b)(s - c)}
\]

```python
def area(base, h1, h2):
    s = (base + h1 + h2) / 2.0
    a = math.sqrt(s * (s - base) * (s - h1) * (s - h2))
    return a
```

13. (8) Assume:

```python
strList = ['goodbye', 'cruel', 'world']
umList = [17, 8, 14]
```

What is output by each of the print statements below?

```python
numList[1] = 11
print strList + numList
goodbye cruel world 11 11 14
numList.append(numList)
print numList
goodbye cruel world 11 11 14 11 14 17
numList.sort()
print numList
goodbye cruel world 11 14 17 11 14 17
numList.extend([165, 43, 22])
print numList
goodbye cruel world 11 14 17 11 14 17 165 43 22
numList.pop()
print numList
[11, 14, 17, 165, 43]
numList.insert(3, 12)
print numList
[11, 14, 17, 12, 165, 43]
numList.reverse()
print numList
[43, 165, 12, 17, 14, 11]```
14. (3) What is the value of List1 after the following code fragment has been executed?

L = ['Always', 'look', 'on', 'the', 'bright', 'side', 'of', 'life']
List1 = [[i.upper(), i.lower()] for i in L if len(i) == 4]

```
[['LOOK', 'look'], ['SIDE', 'side'], ['LIFE', 'life']]
```

15. (2) What does the following code print?

```python
def myFunc(bList):
    bList[0] = 100
    aList = [1, 2, 3]

aList = [5, 6, 7]
myFunc(aList)
print aList
```

16. (5) Assuming the following dictionary:

```python
passwd = {'guido': 'awesome', 'turing': 'genius', 'bill': 'monopoly'}
```

Follow the following code segment. Indicate what is printed on the lines next to each print statement. Make sure the format of your answer is correct.

```python
passwd['newuser'] = 'ImANewbie'
print passwd.keys()  # ['guido', 'turing', 'bill', 'newuser']
print passwd.values()  # ['awesome', 'genius', 'monopoly', 'ImANewbie']
print passwd.items()  # [('guido', 'awesome'), ('turing', 'genius'), ('bill', 'monopoly'), ('newuser', 'ImANewbie')]
print passwd.has_key('bill')  # True

print 'fred' in passwd  # False
```

17. (5) What is the list created from the following list comprehensions?

a) [(x,y) for x in range(2,9,3) for y in range(1,9,3) if x > y]

```
[(2,1), (5,1), (5,4), (8,1), (8,4), (8,7)]
```

b) [ch for ch in 'When can we go home'.lower() if ch not in ['a', 'e', 'i', 'o', 'u', ' ']] (Do NOT include quotes in answer.)

```
[w, h, n, e, n, w, g, h, m]
```
18. (7) Assume the following nonsensical code. Based on the inputs below, what will be the output.

    x, y, z = raw_input('Enter three integer values: ')

    if x == y:
        if x == z:
            print 'Cheesecake'
        elif x > z:
            print 'Hot dogs'
        else:
            print 'Shrimp of any kind'
    elif x > y:
        if x == z:
            print 'Cardigans'
        elif x < z:
            print 'Sweatpants'
        else:
            print 'High heels'
    else:
        if x == z:
            print 'Hawaii'
        elif x > z:
            print 'Aruba'
        else:
            print 'Europe'

a) 5, 5, 5
    Cheesecake
b) 5, 6, 7
    Europe
c) 7, 6, 5
    High heels
d) 7, 6, 8
    Sweatpants
e) 7, 7, 9
    Shrimp of any kind
f) 5, 4, 5
    Cardigans
g) 5, 7, 5
    Hawaii
19. (5) Assume the following class structure. What is output from the following piece of code?

class newClass(object):
    def __init__(self, val, mult = 3):
        self.value = val
        self.mult = mult

    def __str__(self):
        return "The value %d times the multiplier %d is %d" % (self.value, self.mult, self.value * self.mult)

def getVal(self):
    return self.value

def getMult(self):
    return self.mult

def setMult(self, mult):
    self.mult = mult

inst1 = newClass(6,5)
inst2 = newClass(4)

print inst1  # The value 6 times the multiplier 5 is 30
print inst2  # The value 4 times the multiplier 3 is 12
print inst2.getVal() * inst1.getMult()  # 20
print inst1.getVal() * inst2.getMult()  # 18
inst2.setMult(4)
print inst2  # The value 4 times the multiplier 4 is 16

20.(3) Given dictionary D, rewrite this code using exceptions.

    if x in D:
        D[x] += 1
    else:
        D[x] = 1
21. (10) Assume the following program. What is output when it is executed? Put your results in the lines below. Make sure your output is in the proper order (as it would output in the shell).

```python
def first (a, b):
c = a * b + 4
b = c - a
print "First: ", a, b, c
return b

def second (a, b):
c = b + (a * b)
a = b + a - c
print "Second: ", a, b, c
return a

a, b, c = 5, 4, 3
print "Main1: ", a, b, c
b = first (c, a)
c = second(b, c)
print "Main2: ", a, b, c
```

```
Main1: 5 4 3
First: 3 16 19
Second: -32 3 51
Main2: 5 16 -32
```

22. (5) If mySet = set('xyz') and yourSet = set('uvwxyz'), what are the values of: (use correct set notation.

- myset.union(yourset)  
- myset.intersection(yourset)  
- myset.difference(yourset)  
- yourset.difference(myset)  
- myset.symmetric_difference(yourset)

23. (4) Convert 1732 (base 10) to binary and hexadecimal.

Binary: 110110001100

Hexadecimal: 10c4
24. (4) Convert 10110110111 (base 2) to hexadecimal and decimal.

Hexadecimal: 5B7

Decimal: 14163

25. (4) Using an 8-bit byte, show the two's complement representation for 98. Then show how -98 is represented in two's complement. (Again be sure to use 8 bits.)

98 in two's complement: 01100010

-98 in two's complement: 10011110

26. (4) Using 8-bit bytes, perform the following add instructions assuming the number are in two’s complement notation. Note if the answer has a problem with overflow.

\[
\begin{array}{c}
11011001 \\
+11101011 \\
\hline
11001110
\end{array}
\quad
\begin{array}{c}
01100101 \\
+01011110 \\
\hline
10000101
\end{array}
\]

Overflow (Y/N)

Overflow (Y/N)

27. (8) Convert the following number to binary notation and then to floating point notation using the floating point representation discussed in class with 1 bit for the sign bit, 3 bits for the exponent in excess-4 notation and 4 bits for the fraction.

-2 3/4

binary: 10.11

floating-point: \underline{11101011}

5/64

binary: 0.00101

floating-point: 00011010

28. (2) Assuming the same 8-bit represent for floating-point numbers, convert 01011110 from floating point to decimal. Leave your final answer as a decimal fraction.

Decimal: \underline{1\frac{3}{4}}