CMPSCI 119
Spring 2017
Monday, February 27, 2017
Midterm #1 Solution Key
Professor William T. Verts

CMPSCI 119 – Midterm #1 Solution Key – Spring 2017 – Professor William T. Verts

<1> 15 Points – Do any 15; do more for extra credit. The following statements **N=5**, **K=6.7**, **Q="UNIVERSITY OF MASSACHUSETTS"**, **R=[8,[0,1,2],8.25,"FROG",7]**, **P=12**, and **W=(6,2,[4,2],9)** have all been executed. Show the computed result for each statement. Indicate where the computation fails because of some form of error. Be careful about the *type* of the result, particularly integers, floats, long integers, bools, and complexes, and put proper quotes around string results, square brackets around lists, and parentheses around tuples. (+1 each, ignore blank/incorrect answers, -½ for type errors.)

	Question	Answer
1.	N + K	11.7 (float)
2.	P / N	2 (integer)
3.	N * 2	10 (integer)
4.	N * 2.0	10.0 (float)
5.	int(K)	6 (integer, truncate)
6.	int(round(K))	7 (integer, rounded)
7.	N > 3	True (bool)
8.	len(R)	5 (integer)
9.	R[3] + Q[-1]	"FROGS" (string, can use ')
10.	R + W	Error (can’t add list+tuple)
11.	R[1] + W[2]	[0,1,2,4,2] (adding lists)
12.	Q[len(Q)/3]	"Y" (string, Q[9])
13.	N + (R[0] < 12)	6 (integer)
14.	range(N)	[0,1,2,3,4]
15.	range(K)	Either [0,1,2,3,4,5] or Error
16.	range(-2,N)	[-2,-1,0,1,2,3,4]
17.	range(1,P,3)	[1,4,7,10]
18.	[I*I for I in W[2]]	[16,4]
19.	[0 for I in range(P/3)]	[0,0,0,0]
20.	[C for C in R[3]]	["F","R","O","G"] (can use ')

<2> 10 points – Show what is printed by the following code fragment for each given case: (2 points each, no partial credit.)

```
    if (N < 10):
        if (N > 6):
            print "A"
        else:
            print "B"
    elif (N > 20):
        print "C"
    else:
        print "D"
```

Case #1: **N=4**	**"B"**
Case #2: **N=12**	**"D"**
Case #3: **N=30**	**"C"**
Case #4: **N=7**	**"A"**
Case #5: **N=10**	**"D"**

<3> 15 Points – Completely rewrite the following code fragment to perform *exactly* the same task, but using either a **for**-loop or a list comprehension instead of a **while**-loop: (Accept either for-loop or list-comprehension. Remove 10 points for major errors, 5 points for minor errors, 2 points for using different/extra variables.)

```
I = 6                                  L = []
L = []                                 for I in range(6,27,4):
while (I < 27):                            L = L + [I]
    L = L + [I]
    I = I + 4                          L = [I for I in range(6,27,4)]
```

<4> 20 Points – Show what is printed out as the result from calling **Main()** (four lines total): (5 points each line.)

```
def F1 (M,J,Q):
    R = J - Q
    return M + R

def F2(Z,Q,J=5):
    return F1(Q,Z,J)

def F3(R,Z,Q):
    return F2(Q,Z+2,R)

def Main():
    print F1(5,2,6)               1
    print F3(1,6,3)               10
    print F2(9,3)                 7
    print F2(2,6,2)               6
    return
```

<5> 15 Points – The following code contains errors. Locate each one and indicate what the correction(s) should be. Don't rewrite any code statements; just correct the mistakes. (I count 7 errors for 2 points per error found, 1 free point. Remove -1 for misidentifying correct code as errors.)

```
def SigNum(N):
    If (N < 0): Result = -1              If → if
    elseif (N > 0): Result = 1           elseif → elif
    else: Result = 0                     missing colon
    return result                        r → R

def PrintAnswer(N):                      missing colon
    X = float(Signum(N))                 Signum → SigNum
    print "The answer is ', X            ' → "
    return

def Main():
    PrintAnswer(input("Enter a number "))    missing )
    return
```

<6> 25 Points – The following code loads in a graphic picture from a file. Finish the function to transform the picture by first exchanging the red and blue values of each pixel (10 points), and then by making the green value of each pixel go to 255 if it is greater than 160 but leaving it unchanged otherwise (10 points). Your solution should take only 7-8 lines of code. (Example: **R**=23, **G**=192, **B**=200 goes to **R**=200, **G**=255, **B**=23.) (In each section, remove 5 points for minor logic errors, all 10 points for major logic errors. Reserve the remaining 5 points for syntax/indentation/capitalization errors: 1 point each.)

```
def Main():
    Filename = pickAFile()
    Canvas = makePicture(Filename)
    show(Canvas)
    for Y in range(getHeight(Canvas)):
        for X in range(getWidth(Canvas)):
            PX = getPixel(Canvas,X,Y)
            R = getRed(PX)
            B = getBlue(PX)
            setRed(PX,B)
            setBlue(PX,R)

            G = getGreen(PX)
            if G > 160:
                setGreen(PX,255)           |
        repaint(Canvas)
    return
```