COSC 1336 Homework 5

Relevant reading: Chapters 5 & 6

Due: Oct. 25, 2:30 pm

(Late date: Nov. 1, 2:30 pm)

40 Points

More on loops

Problem 1. [**14 points**] For each of the following, desk-check the code, showing the values of each variable as the code executes. Also, determine *exactly* what will appear on the screen when the code is executed to completion. You will get very little credit for a correct final answer if it is not accompanied by a legible desk check.

a. [**3 points**]

```
x = 20
while (x < 20) and (x >= 0):
    print(x, end=" ")
    x = x - 4
```

b. [**3 points**] Note that this loop is the same as the one in part a, except that the exit condition is tested at the *end* of the loop rather than the beginning. That is, this is a post-test loop, whereas the previous one is a pre-test loop.

```
x = 20
while True:
    print(x, end=" ")
    x = x - 4
    if not (x < 20) and (x >= 0):
        break
```

c. [**4 points**]

```
k = 1
for i in range(5):
    for j in range(i+1):
        print(format(k, "2d"), end=" ")
        k += 1
    print()
```

d. [**4 points**]

```
sum = 0
y = 0
while True:
    for i in range(4):
        sum += i
    y += 2
    y = y + 2;
    if y >= 6:
        break
print("Sum:",sum)
```

Value-returning functions

Problem 2. [**12 points**] **Short answer**

a. [**2 points**] Write a single Python statement that generates a random number in the range of 1 through 100 and assigns it to a variable named `rand`.

b. [**2 points**] Write a single Python statement that generates a random number from the sequence `5, 10, 15, 20, 25, 30` and assigns it to a variable named `num`.

c. [**2 points**] A program contains the following function definition:

```
def cube(num):
    return num * num * num
```

Write a statement that passes the value 4 to this function and assigns its return value to the variable `result`.

d. **[3 points]** Write a function called `times_ten` that accepts a number as an argument and returns the value of the argument times 10.

e. **[3 points]** Write a function called `is_odd` that accepts an integer as an argument and returns a Boolean indicating whether or not the input argument is odd. Try to write this function without using a conditional, but you will get most of the credit if you do use a conditional.

Problem 3. [14 points] For each of the following, create a stack diagram that shows what happens when the `main` function is run to completion. The diagram should include the stack frames for function calls that have returned.

a. **[2 points]**

```
def square(x):
    xSq = x * x
    return xSq

def main():
    num = 5
    num = square(num)
    print("Answer:", num)
```

b. **[3 points]**

```
def square(x):
    x = x * x
    return x

def main():
    x = 5
    xSq = square(x)
    print("Answer:", xSq)
```

c. **[4 points]**

```
def func(num):
    num = num % 5
    return num == 0

def main():
    num = 30
    b = func(num)
    if b:
        print(num / 5)
    else:
        print("no")
```

d. **[5 points]**

```
def hypotenuse(a, b):
    aSq = a * a
    bSq = b * b
    return sqrt(aSq + bSq)

def distance(x1, y1, x2, y2):
    d = hypotenuse(x2 - x1, y2 - y1)
    return d

def main():
    x1, y1 = 0, 0
    x2, y2 = 3, 4
    dist = distance(x1, y1, x2, y2)
    print("Distance:",dist)
```