Super Quiz

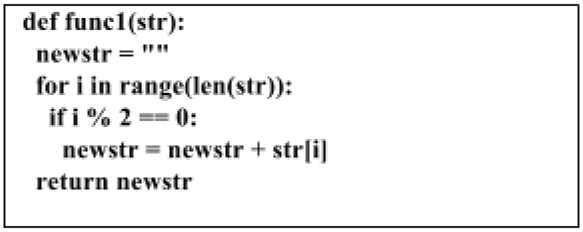
Name ____

Write your answers to the multiple choice questions here.

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____
10. _____
11. _____
12. _____
13. _____
14. _____
15. _____
16. _____
17. _____
18. _____
19. _____
20. _____

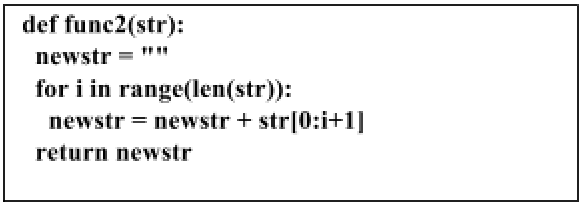
Multiple Choice Questions (2 points each, for 40 points total)

1. What does **func1('CS1400')** return?
   a) CS14
   b) C10
   c) S40
   d) CS1400



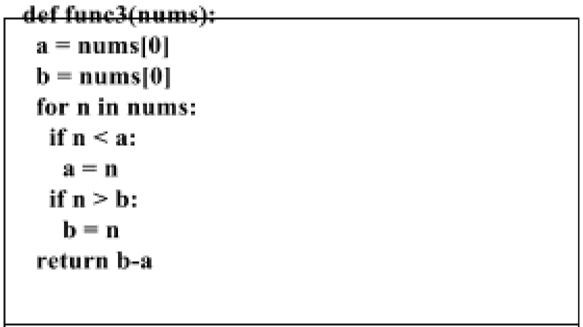
```
def func1(str):
  newstr = ""
  for i in range(len(str)):
    if i % 2 == 0:
      newstr = newstr + str[i]
  return newstr
```

2. What does **func2("Code")** return?
   a) Code
   b) CCooddee
   c) CCoCodCode
   d) CodeodedeC



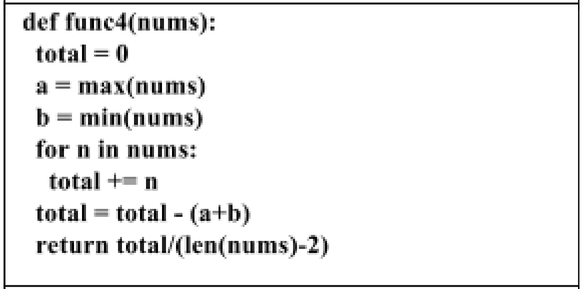
```
def func2(str):
  newstr = ""
  for i in range(len(str)):
    newstr = newstr + str[0:i+1]
  return newstr
```

3. What will **func3([5,4,2,1,8,3,5,9,7,2])** return?
   a) 10
   b) 9
   c) 8
   d) 7



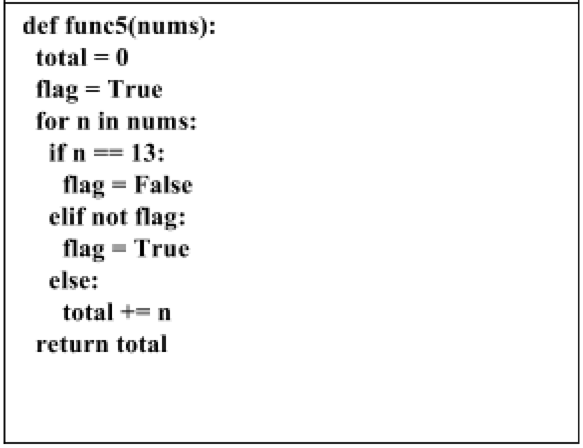
```
def func3(nums):
  a = nums[0]
  b = nums[0]
  for n in nums:
    if n < a:
      a = n
    if n > b:
      b = n
  return b-a
```

4. What does **func4([2,1,5,5,10,6,7])** return? **max(list)** returns the largest value in list. **min(list)** returns the smallest value in list.
   a) 1
   b) 5
   c) 7
   d) 10



```
def func4(nums):
  total = 0
  a = max(nums)
  b = min(nums)
  for n in nums:
    total += n
  total = total - (a+b)
  return total/(len(nums)-2)
```

5. What does **func5([1,3,13,10,5])** return?
   a) 9
   b) 15
   c) 19
   d) 32



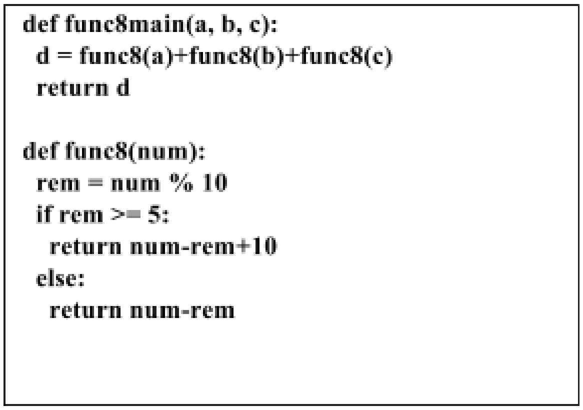
```
def func5(nums):
  total = 0
  flag = True
  for n in nums:
    if n == 13:
      flag = False
    elif not flag:
      flag = True
    else:
      total += n
  return total
```

6. What does **func6main(13,15,17)** return?
   a) 13

b) 15
c) 17
d) 45

7. How many times is **func6** called by **func6main**?
a) 1
b) 2
c) 3
d) 0

8. What does **func8main(12,15,18)** return?
a) 15
b) 20
c) 35
d) 50

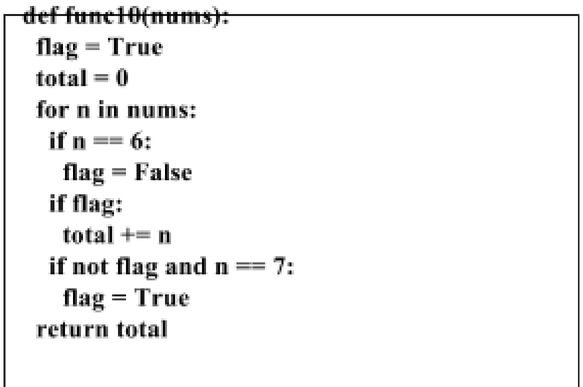


```
def func8main(a, b, c):
  d = func8(a)+func8(b)+func8(c)
  return d

def func8(num):
  rem = num % 10
  if rem >= 5:
    return num-rem+10
  else:
    return num-rem
```

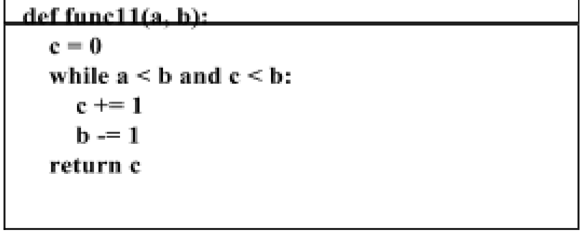
9. What does **func8(12)** return?
a) nothing
b) 10
c) 12
d) 20

10. What does **func10([1,2,2,6,99,99,7,3])** return?
a) 8
b) 11
c) 12
d) 219



```
def func10(nums):
  flag = True
  total = 0
  for n in nums:
    if n == 6:
      flag = False
    if flag:
      total += n
    if not flag and n == 7:
      flag = True
  return total
```

11.



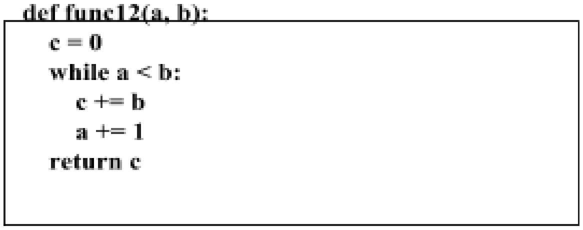
```
def func11(a, b):
    c = 0
    while a < b and c < b:
        c += 1
        b -= 1
    return c
```

What does **func11(3, 7)** return?

a) 3
b) 4
c) 7
d) 10

12. What does **func12(2, 6)** return?

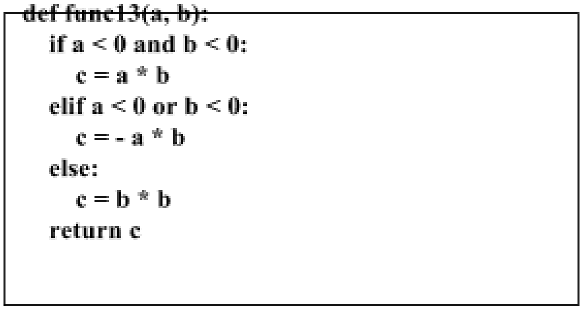
a) 8
b) 12
c) 16
d) 24



```
def func12(a, b):
    c = 0
    while a < b:
        c += b
        a += 1
    return c
```

13. What does **func13(4, -8)** return?

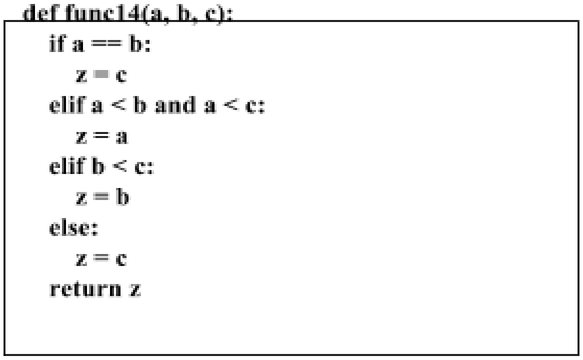
a) -32
b) 0
c) 32
d) 64



```
def func13(a, b):
    if a < 0 and b < 0:
        c = a * b
    elif a < 0 or b < 0:
        c = - a * b
    else:
        c = b * b
    return c
```

14. What does **func14(4, -8, -6)** return?

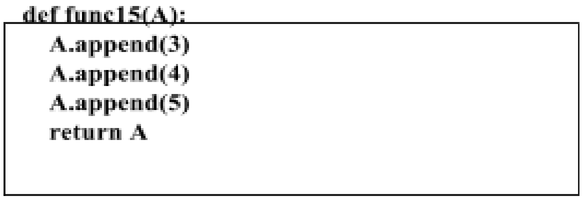
a) -8
b) -6
c) 4
d) 48



```
def func14(a, b, c):
    if a == b:
        z = c
    elif a < b and a < c:
        z = a
    elif b < c:
        z = b
    else:
        z = c
    return z
```

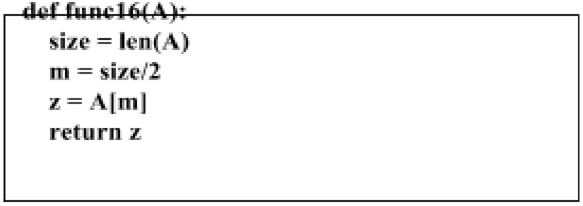
15. What does **func15([6,7,8])** return?

a) [3, 4, 5, 6, 7, 8]
b) [5, 4, 3, 6, 7, 8]
c) [6, 7, 8, 5, 4, 3]
d) [6, 7, 8, 3, 4, 5]



```
def func15(A):
    A.append(3)
    A.append(4)
    A.append(5)
    return A
```

16.



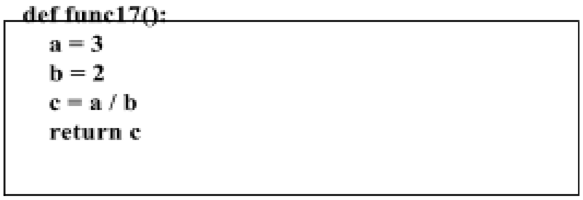
```
def func16(A):
    size = len(A)
    m = size/2
    z = A[m]
    return z
```

What does **func16([3,5,7,9,11])** return?

a) 3
b) 11
c) 9
d) 7

17. What does **func17()** return?

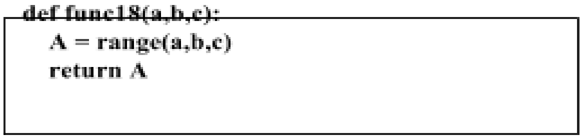
a) 3/2
b) 1
c) 1.0
d) 1.5



```
def func17():
   a = 3
   b = 2
   c = a / b
   return c
```

18. What does **func18(4, 10, 3)** return?

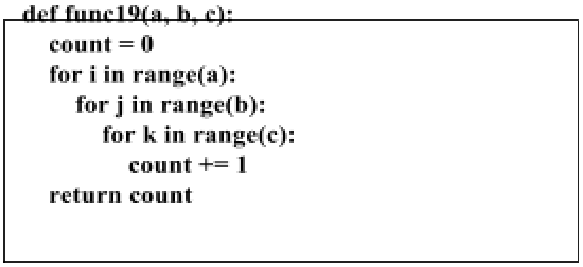
a) [4, 7]
b) [3, 7, 10]
c) [4, 5, 6, 7, 8, 9, 10]
d) [4, 5, 6, 7, 8, 9]



```
def func18(a,b,c):
   A = range(a,b,c)
   return A
```

19. What does **func19(5,4,3)** return?

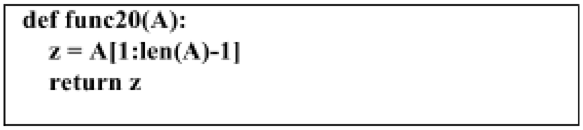
a) 20
b) 40
c) 60
d) 80



```
def func19(a, b, c):
   count = 0
   for i in range(a):
      for j in range(b):
         for k in range(c):
            count += 1
   return count
```

20. What does **func20([8,3,-1,1,2])** return?

a) [8, 3, -1, 1, 2]
b) [8, 3, -1, 2]
c) [8, 3, 1, 2]
d) [3, -1, 1]



```
def func20(A):
   z = A[1:len(A)-1]
   return z
```

Write the Code Questions (2 points each, for 20 points total)

21. Write a function named swap that receives 2 parameters, and returns the two parameters in reverse order.

```
Example calls:
a,b = swap(3, 5)
print a, b

Prints:
5 3
```

22. Write a function called slice_of_perfection that receives 3 parameters, a string and two numbers. The function returns a character string starting at the character with index specified by the second parameter, of length specified by the third parameter. The input string will always be long enough.

```
Example calls:
a = slice_of_perfection("memphis", 2, 3)
b = slice_of_perfection("pennsylvania", 6, 4)
c = slice_of_perfection("rome", 0, 4)
print a, b, c

Prints:
mph lvan rome
```

23. Write a function called grow_one that receives two parameters, one is a list and the other is a single item. This function returns the list with the item added onto the end of it.

```
Example calls:
a = grow_one([1, 2, 3], 4)
b = grow_one([6, 8], 7)
print a, b
```

```
Prints:
[1, 2, 3, 4] [6, 8, 7]
```

24. Write a function named total_slice, that receives 2 numeric parameters, and returns the sum of all numbers from the first parameter to 1 less than the second parameter.

```
Example calls:
a = total_slice(1, 4)
b = total_slice(5, 8)
c = total_slice(8, 5)
print a, b, c

Prints:
6 18 0
```

25. Write a function named xor, that receives 2 boolean parameters, and returns True if one parameter is True and the other is False. Otherwise, it returns False.

```
Example calls:
a = xor(False, True)
b = xor(True, True)
c = xor(True, False)
print a, b, c

Prints:
True False True
```

26. Write a function named centered_average, that receives a list of integers as a parameter. It returns the "centered" average of the numbers, which we'll say is the mean average of the values, except ignoring the largest and smallest values in the array. If there are multiple copies of the smallest value, ignore just one copy, and likewise for the largest value. Use int division to produce the final average. You may assume that the array is length 3 or more.

```
Example calls:
a = centered_average([1, 2, 3, 4, 100])
b = centered_average([1, 1, 5, 5, 10, 8, 7])
c = centered_average([-1, -4, -2, -4, -2, 0])
print a, b, c

Prints:
3 5 -3
```

27. Write a function called parrot_trouble, that receives two parameters. The first is boolean (True if our parrot is talking); the second is an integer, the hour in military time (0..23). We are in trouble if the parrot is talking and the hour is before 7 or after 20. Return True if we are in trouble.

```
Example calls:
a = parrot_trouble(True, 6)
b = parrot_trouble(True, 7)
c = parrot_trouble(False, 6)
print a, b, c

Prints:
True False False
```

28. Write a function called list_total2 that receives a list of integers as a parameter, and returns the total of all even numbers in the list.

```
Example calls:
a = list_total2([1, 2, 3, 4, 100])
b = list_total2([1, 1, 5, 5, 10, 8, 7])
c = list_total2([-1, -4, -2, -4, -2, 0])
print a, b, c

Prints:
106 18 -12
```

29. Write a function named power that receives 2 parameters, and returns the result of raising the first number to the power of the second.

```
Example calls:
a = power(3, 2)
b = power(2, 3)
print a, b

Prints:
9 8
```

30. You are driving a little too fast, and a police officer stops you. Write a function named caught_speeding that receives two parameters, a number representing your driving speed, and a boolean representing whether it is your birthday or not. The function returns an int value: 0=no ticket, 1=small ticket, 2=big ticket. If speed is 60 or less, the result is 0. If speed is between 61 and 80 inclusive, the result is 1. If speed is 81 or more, the result is 2. Unless it is your birthday -- on that day, your speed can be 5 higher in all cases.

Example calls:

```
a = caught_speeding(60, False)
b = caught_speeding(65, False)
c = caught_speeding(65, True)
print a, b, c
```

Prints:

```
0 1 0
```