Topic list

Key: SA = short answer, CR = code reading, FR = free response, CW = code writing

Note 1: any topic listed at CW or FR rank may be tested at all ranks.
Note 2: any topic listed at the CR rank may also be tested at the SA rank.

Algorithms and Abstraction (SA, FR)

1. Give a high level algorithm for printing out a list in sorted order.

2. Give a high level algorithm for returning the sum of every other element in a list.

Programming Basics (CW)

1. Write a Python program that prints an item and its corresponding type.

2. Output the results of the following statements:
   a. `float(int(32.58))`
   b. `type(7//2)`
   c. `type("01151")`

How Python Works (CR)

1. What is the job of the interpreter?

2. What type of error is each of the following?
   a.
   ```
   x = 5
   x = x + y
   ```
   b.
   ```
   if x = 2:
     print ("Hello")
   ```
   c.
   ```
   x = 0
   x = x + "no"
   ```

Functions (FR, CW, SA)

1. If we have the the following function:

   ```
   def summation(a,b):
       print(a+b)
   c = summation(2,4)
   ```

   What will c be equal to after we call this function? If there is an error, fix and explain it.

2. What does the following function returns?

```
def f(x):
        x + 42

print(f(5))
```

Data Representation (SA, FR)

1. If we only had 5 bits to use, what is the minimum and maximum number we can represent using 5 bits?

2. Convert the following decimal numbers into their binary representation using only 4 bits. If there aren't enough bits then only represent the lower 4 bits: 0, 17, 23, 5, 8, 2.

3. Explain the difference in the simple approach and actual approach in the binary representation of negative numbers.

Booleans and Conditionals (CW)

1.
```
def f(x, y, z):
    result = ""
    if (x + y) % 2 == 0:
        result += str(x)
    if (y + z) % 2 == 1:
        result = str(y) + result
    if z % 4 == 3:
        result = ""
    return result

print(f(1, -7, 526), f(8, 43, 2), f(9, 101, 11))
```

2. Write a function to determine whether somebody should eat ice cream on a hot day based on temp (must be greater than 60 degrees) and hunger (must be greater than 0.5)

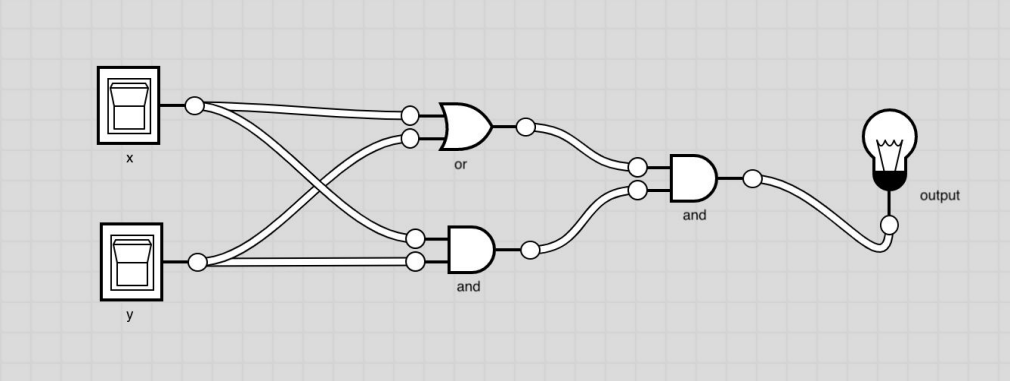
3. What is the difference between the "and" vs. "or" operations in terms of their relationship with the boolean True?

Topic list

Circuits and Gates (FR, SA)

1. How does a half adder work? How does a full adder work? What are the differences?

2. What boolean operation does the following logic circuit behave like?



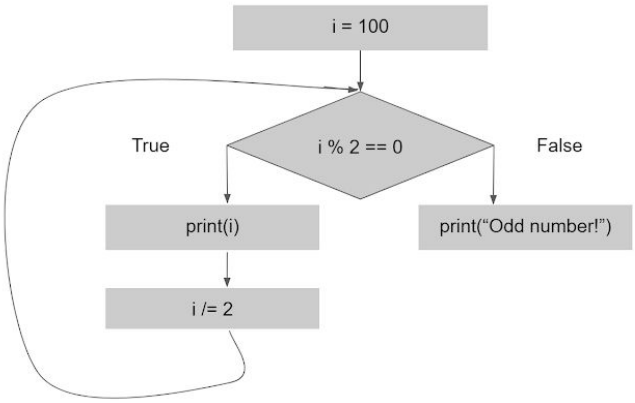
3. What is the purpose of C_in and C_out in a full adder?

While Loops (CW, FR)

1. Write the function `createTriangle(n)` to recreate the following pattern with a while loop given n number of rows.

```
print(createTriangle(3))
*
**
*
```

2. Write the while loop that corresponds with this flow chart.



3. Use while loop to write function `hasConsecutiveDigits(n)` that takes in a possibly-negative int value n and returns True if that number contains two consecutive digits that are the same, False otherwise.

4. Write the function `isPowerFour(n)` that takes in a number n and returns True if n is a power of 4, returns False otherwise.

Testing and Debugging (FR, CR, SA)

1. List 5 categories of test cases, and give an example for each

2. Indicate if there's anything wrong with the following statements/functions:
   a) Kevin wrote a function that takes in a number n and returns the number of multiples of 3 up to that number.

```
def f(n):
    count = 0
    number = 1
    while (number < n):
        if (number % 3 == 0):
            count = count -1
   return count
```

   b) Zack wrote this function called `same(s)` trying to count the number of pairs of the same character inside a string. (for example: `same("dad")` returns `1`)

```
def same(s):
    counter = 0
         for i in range (len(s)-1):
              for j in range (1, len(s)):
                   if (s[i] == s[j]):
                        counter = counter + 1
    return counter
```

For Loops (CW, FR)

1. Explain when you might use a for-range loop and when you might use a for-each loop.

2. Similarly, when would you use a while loop versus a for loop? Can you always convert a for loop to a while loop? Can you always convert a while loop to a for loop?

3. Write a function `numberOfFactors(n)` which takes in a natural number (not including 0) and returns the number of factors it has.

4. Using a for loop, write the function `fizzBuzz(n)` that prints every number from 0 to n-1 inclusive. If the number is divisible by 3, print “fizz” instead of the number. If the number is divisible by 5, print “Buzz” instead of the number. If divisible by both 3 and 5, print “fizzBuzz” instead of the number.

5. Using a for loop, write the function `sumAllEven(n)` that finds the sum of all even numbers less than or equal to n.

Strings (CW, CR)

1. Write a function `reverseString(s)` that returns the string s reversed.

2. What would the following code print?

```
def mystery(s, n):
    for word in s.split(" "):
        if len(word) == n:
            return word
    return "Darn!"

print("She sells seashells down by the seashore", 4)
```