

15-110: Principles of Computing

Sample Exam #1

The following is a "sample exam" that you can use to practice after you have studied for the exam. Keep in mind that the actual exam will have its own questions, which may not match these questions closely, but they will cover similar topics at a similar level of depth. The exam is 50 minutes in length. Read each question carefully.

1. (13 pts)

- a. (3 pts) Put the following computing inventions in order from earliest invention to latest invention. Use the letters to indicate the order.

- A. Difference Engine
 - B. ENIAC
 - C. Hollerith's Tabulating Machine
 - D. Pascaline
 - E. Apple Macintosh
-

- b. (4 pts) Complete the following byte measurements.

i. 32GB = 2 bytes

ii. 2^{23} bytes = 8 _____

- c. (3 pts) According to Moore's Law, if the computational power of a computer doubles every 2 years, a computer 16 years from now will be how many more times powerful than today's computer?

- d. (3 pts) When you print a report using a laser printer, can you assume that no one can tell who printed it? Give a yes/no answer followed by a one sentence explanation.

2. (30 pts) Expressions and basic functions

- a. (12 pts) For each of the following Python expressions, write down the value that would be output if the expression was evaluated using a python3 interpreter.

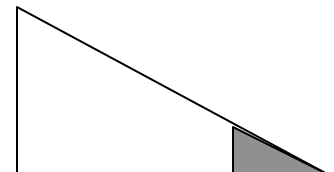
<code>40 // 9</code>	_____
<code>2 * 2 ** 4</code>	_____
<code>6 + 4 * 2 - 1</code>	_____
<code>15 / 2</code>	_____
<code>15 % 2</code>	_____
<code>2 != 2</code>	_____

- b. (4 pts) Write a Python3 function `triangle_area` that takes two parameters `b` and `h`, respectively, for the base and height of a triangle, and returns the area of the triangle given by the formula: $\text{area} = \frac{1}{2} (\text{base} \times \text{height})$.

```
def triangle_area(b,h):
```

- c. (4 pts) Write a Python3 function `truncated_triangle` that takes two parameters, `b` and `h`, representing the base and height of a triangle and returns the area of the triangle with the tip cut off as shown in the diagram. (In the picture, you would return the area of the white section.) The tip is also a triangle whose height and base are 25% of the height and base of the larger triangle. Use the `triangle_area` function in this function to compute your answer. Do not use the area formula directly.

```
def truncated_triangle(b,h):
```



- d. (2 pts) The Python3 function below prints a sequence of numbers. Which of the following is the output of the function expressed in terms of m and n ?

```
def mystery1(m,n):  
    i = 0  
    while i <= n-1:  
        i = i + 1  
        print(i ** m, end = "  ")
```

- A. $1 \ 2^m \ 3^m \ \dots \ n^m$
B. $1 \ 2^n \ 3^n \ \dots \ m^n$
C. $1 \ m^2 \ m^3 \ \dots \ m^n$
D. $1 \ n^2 \ n^3 \ \dots \ n^m$
-

- e. (4 pts) If the print statement were taken outside of the while loop as shown below, what would the function call `mystery2(2,3)` print?

```
def mystery2(m,n):  
    i = 0  
    while i <= n-1:  
        i = i + 1  
    print(i ** m, end = "  ")
```

- f. (4 pts) Consider the following Python function:

```
def mystery3(m,n):  
    i = 0  
    result = 0  
    while i <= n-1:  
        i = i + 1  
        result = result + i**m  
    return result
```

What would the value of the variable `x` be after executing the statement below?

```
x = mystery3(2,4)
```

HINT: Trace the function keeping track of `i` and `result` for each iteration of the loop.

3. (25 pts)

a. (8 pts) Assume the following list definition in Python:

```
a = [1, 2, 4, 8, 16, 32, 64, 128, 256, 512]
```

What would be displayed using python3 for each of the following expressions?

`len(a)` _____

`a[2]` _____

`a[1:4]` _____

`64 in a` _____

b. (12 pts) Complete the function `sum_list` below that takes a number list as its parameter and returns the sum of the numbers in the list.

```
def sum_list(numlist):  
    index = _____  
    sum = _____  
    while _____:  
        sum = sum + _____  
        index = index + _____  
    return sum
```

c. (5 pts) Write an equivalent function that uses a `for` loop and does not use an index variable. That is, you cannot write `numlist[index]`.

4. (20 pts) A list is defined to be "out of order" if it is not in ascending order (no duplicates). We wish to write a function `out_of_order` that returns the first value that is "out of order", that is, the first item that is not in ascending order. The function should return `None` if the list is entirely in ascending order (i.e. sorted). For example, `out_of_order([1,5,17,12,24])` should return 12 since that is the first value that is out of order in the list.

a. (10 pts) Here is an iterative algorithm that computes the result:

1. Set *index* equal to the first index of the list.
2. While *index*+1 is valid, do the following:
 - a. If the list item at *index* is greater than the list item at *index*+1, then return the list item at *index*+1 (which is the one out of order)
 - b. Add 1 to *index*.
3. Return `None`.

Complete the function below using the iterative algorithm above:

```
def out_of_order(numlist):
```

b. (10 pts) Consider the following recursive algorithm for this problem:

1. If the list has fewer than two elements, return `None`.
2. Otherwise if the first element of the list is greater than the second element, return the second element.
3. Otherwise return the result using the same function on the list without the first element.

Complete the function below using the recursive algorithm above:

```
def out_of_order(numlist):  
    if _____ < 2:  
        return None  
  
    elif _____ > _____:  
        return _____  
    else:  
        return _____
```

5. (12 pts) The following questions deal with evaluating the efficiency of a computation as a function of the size of the data.

a. A machine paints a triangle of stars as shown below for $n=5$:

```
*  
**  
***  
****  
*****
```

What is the worst-case order of complexity of this operation expressed using big O notation for a triangle with height n ? _____

b. Suppose it takes 2 minutes to paint the triangle of size n . If we double the size of the triangle to $2n$, how long will it take approximately to paint the triangle, in minutes? _____

c. What is the worst-case order of complexity for the `out_of_order` function of the last problem if the list has n elements? Express your answer using big O notation. _____
(HINT: How many comparisons does the iterative algorithm need to do in the worst-case?)

d. Give an example of a list of size 6 that exhibits the worst case behavior.

e. What is the best-case order of complexity of the same `out_of_order` function if the list has n elements using big O notation? _____

f. Give an example of a list of size 6 that exhibits the best case behavior.
