GitHub Username:

MY470 Computer Programming

Mock Problem Set 5, AT 2023

Practicing order of growth analysis for time complexity

This assessment takes the form of a more traditional problem set, where each problem stands by itself and is unrelated to the others.

You are expected to complete the problems on your own.

You have 25 minutes to complete the problem set.

Instructions for Problems 1–5

Give the order of growth for the function and explain your reasoning in a couple of sentences.

Problem 1

In [1]:

```
def sum_array(array):
    """Calculate the sum of all elements in a 3-dimensional array
    (list of lists of lists)."""

    total = 0
    for layer in array:
        for row in layer:
            for element in row:
                total += element
    return total

# Your answer:
#
#
#
#
#
#
#
#
#
```

Problem 2

In [ ]:

```
def is_power_of_two(n):
    """Check if an integer is a power of two.
    (i.e., 1, 2, 4, 8, 16, 32, 64, etc.)"""

    while n > 1:
        n = n / 2

    if n == 1:
        return True
    else:
        return False
# Your answer:
#
#
#
#
#
#
#
```

Problem 3

In [ ]:

```python
def is_sorted(int_list):
    """Check if a list of ints is sorted."""
    sorted_list = sorted(int_list)
    if sorted_list == int_list:
        return True
    return False
# Your answer:
#
#
#
#
#
#
```

Problem 4

In [1]:

```python
def is_sorted2(string_list):
    """Check if a list of strings is sorted
    (different approach to P3)."""
    for i in range(len(string_list) - 1):
        if string_list[i] > string_list[i + 1]:
            return False
    return True
# Your answer:
#
#
#
#
#
#
```

Problem 5

In [2]:

```python
def sum_nd_array(ndarray):
    """Calculate the sum of all elements in a numeric
    n-dimensional array. Each dimension is of the same size "d"."""
    if isinstance(ndarray, (int, float)):
        return ndarray
    total = 0
    for item in ndarray:
        total += sum_nd_array(item)
    return total
# Your answer:
#
#
#
#
#
#
#
```

Instructions for Problems 6–7

Write the function as described in the docstring and called under. Then, give the order of growth for the function and explain your reasoning in a couple of sentences.

Problem 6

In [3]:

```
def fizz(n):
    """Print out all numbers from 1 to n (inclusive),
    replacing multiples of 7 with "Fizz"."""
    # complete function here













fizz(16)
# Your answer for order of growth:
#
#
#
#
#
#
#
#
#
#
```

```
1
2
3
4
5
6
Fizz
8
9
10
11
12
13
Fizz
15
16
```

Problem 7

In [1]:

```
def three_char_words(l1, l2, l3):
    """Given three lists, first identify the strings in each
    list. Assume all strings are a single character and return
    a list of all possible three-character "words" that can be
    made from combining one character from each list (allowing
    for repeats). You will need to remove any non-strings
    (ints, floats, etc.) in advance, but do not alter the
    case of any letters or remove non-letters.
    """


























l1 = ['a']
l2 = ['?', 1, 'C']
l3 = ['c', 'a']
ls = three_char_words(l1, l2, l3)
print(ls)
# Your answer for order of growth:
#
#
#
#
#
#
#
#
#
```

```
['a?c', 'ac?', '?ac', '?ca', 'ca?', 'c?a', 'a?a', 'aa?', '?aa', '?
aa', 'aa?', 'a?a', 'aCc', 'acC', 'Cac', 'Cca', 'caC', 'cCa', 'aCa
', 'aaC', 'Caa', 'Caa', 'aaC', 'aCa']
```