Lab 4 Manual

Lab 4 Outline

Part 1: Experimenting with Python [35 mins]

- *First occurrence*
- *Debugging sum of every kth number in a range*

Part 2: Quiz [15 mins]

Now that you have completed the lab, you will take a short mandatory quiz on the lab material. To take the quiz, **your lab leader will provide you with the link and passcode to access the Lab Quiz on your Ed account**. This will be a **timed (15 mins)**, multiple choice quiz on the lab material. If you completed the lab, you will be able to answer all of the questions on the quiz.

You are required to take the quiz during your synchronous lab session. Make sure to leave enough time to take the quiz before the end of the lab session.

First Occurrence

Create a new file `first_occurrence.py`. Implement a function `first_occurrence(nums, threshold)` which takes in a list `nums` and a number `threshold.` The elements of `nums` are sorted in ascending order, so `nums[0] <= nums[1], nums[1] <= nums[2]`, and so on.Your function needs to return **the index** of **the first** number in `nums` which is **greater than or equal to** `threshold`. If none of the numbers in `nums` satisfies the requirement, return an integer `-1`.

In this example, what test cases would you like to consider to ensure the correctness of your code? Here are some examples for you to think of as well as what your function should return in those scenarios, and feel free to ask your TA if you've got some other ideas!

- `nums` is an empty list
  - Example: `first_occurrence([], 2)` should return `-1`
- When `nums` have multiple occurrences of `threshold`
  - Example: `first_occurrence([1,2,2,5,8], 2)` should return `1`
- When all of the numbers in `nums` are smaller than `threshold`
  - Example: `first_occurrence([1,2,2,5,8], 100)` should return `-1`
- Other examples:
  - `first_occurrence([1,2,2,5,8], 4)`should return `3`

Debugging Sum of Every Kth Number

Copy and paste the following code into a new file `sum_every_kth.py`:

```python
def sum_every_kth(start, end, k):
    if k >=0 or start <= end:
        return None

    total = 1
    for number in range(start, end, k-1)
    total = number
```

The function is meant to compute the sum of every `k`th number (**k<0**) from `start` (**inclusive**) to `end` (**exclusive**). You can assume that the parameters `start`, `end`, `k` are all integers.

When `start`, `end` and `k` do not form a valid range, i.e. the range contains no element or it violates the problem setting, the function returns `None`.

1. Fix the bugs of the code.

2. Try out different test cases. For example:

- `sum_every_kth(6, 3, -1)` should return 15
- `sum_every_kth(6, 2, -1)` should return 18
- `sum_every_kth(6, 4, -2)` should return 6
- `sum_every_kth(0, 2, -1)` should return None
- `sum_every_kth(1, 6, 2)` should return None