Python Programming

Practice

Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)

Practice: Find most frequent number

- Read a line of N integers. Each integer is 0 <= value <= 150
- Find the value that repeated the most number of times.
  - If there are many solutions: find the **smallest** value
- Input ⇒ output
  - 1 2 1 3 1 5 5 ⇒ Value 1 repeated 3
    - Observe:
    - 1 repeated 3 times: the largest
    - 2 repeated 1 time
    - 5 repeated 2 times
  - 5 5 5 5 2 3 3 3 3 ⇒ Value 3 repeated 4
- Stop video and think
  - Do it with nested loops
  - Can you do it with a single loop?

Practice: Find most frequent number

- Nested loop solution is to loop every item and then count how many times
- **Observe**: this is a nested loop solution
  - Count function makes a loop
- Overall: O(n^2)
- Better?
  - Hint: value 0 to 150 only

```
3   def most_frequent_slow(lst):
4       most_value, frequency = None, 0
5       for value in lst:
6           cnt = lst.count(value)
7           if cnt > frequency:
8               most_value, frequency = value, cnt
9           elif cnt == frequency:
10              most_value = min(most_value, value)
11      return most_value, frequency
12
13
14  if __name__ == '__main__':
15      lst = list(map(int, input().split()))
16      most_value, frequency = most_frequent_slow(lst)
17      print('Value', most_value, 'repeated', frequency)
18
```

Practice: Find most frequent number: Faster

- Let’s use another list
- For now: think of the word array = list
- We will use a trick called **frequency array**
  - Think of each value as an index
    - If we have value 50 ⇒ lst[50]
  - If we have M values, create list of M+1 values
- Iterate on the list and increment each time lst[value]
- Now the array represents the frequency of each value!
- Find the argmax!
- Try to code it!

Practice: Find most frequent number: Faster

- Surprisingly: the most efficient code is shorter and easier to read!
- This code do 2 linear loops
  - The first is N steps
  - The second is max 150 steps

```
2   def most_frequent_fast(lst):
3       freq_lst = [0] * (max(lst)+1)
4
5       for value in lst:
6           freq_lst[value] += 1
7
8       # argmax - Observe: the tie is also handled!
9       most_value = freq_lst.index(max(freq_lst))
10
11      return most_value, freq_lst[most_value]
12
```

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."