

Programming

1D Arrays Homework 3

Mostafa S. Ibrahim

Teaching, Training and Coaching for 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)



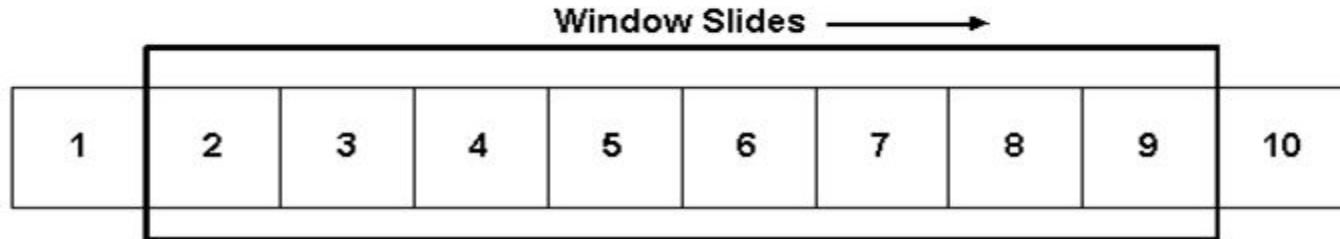
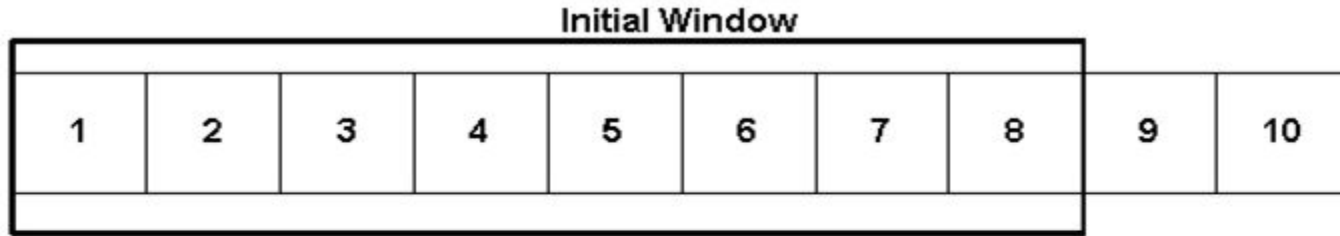
Materials have **COPYRIGHTS** - Can't use without direct **PERMISSION**

Problem #1: Recamán's Sequence

- The first terms of this sequence are 0, 1, 3, 6, 2, **7**, ...
 - So, looking at the **value** 7, its **index** is 5 (in a zero-based array)
 - The next value is either:
 - **LastValue - LastIndex - 1** if the following 2 conditions are satisfied:
 - 1) value > 0 2) It hasn't already appeared within the sequence
 - e.g. 7 (last value)-last index (5)-1 = 7-5-1 = 1 (> 0, but is already in the sequence)
 - Or **LastValue + LastIndex + 1** = **7+5+1 = 13**
- Read in a zero-based integer index ([1, 200]), and print this index's value
 - e.g. (6 ⇒ 13), (9 ⇒ 21), (17 ⇒ 25)
- Don't use nested loops
- The series is: 0, 1, 3, 6, 2, 7, **13**, 20, 12, **21**, 11, 22, 10, 23, 9, 24, 8, **25**, 43

Background: Fixed Sliding Window

- Indicates a group of consecutive/contiguous numbers. Can be either of fixed or variable size
 - You 'slide' to the next window in the array



Background: Fixed Sliding Window

- Assume we have a list: **1 0 3** -4 2 -6 9
- Let's imagine our sliding window (sublist) size is 3
- Let's print out all windows of length 3, and their sum
 - 1 0 3 \Rightarrow sum = 4
 - 0 3 -4 \Rightarrow sum = -1 [observe that 0 and 3 are common]
 - 3 -4 2 \Rightarrow sum = 1
 - -4 2 -6 \Rightarrow sum = -8
 - 2 -6 9 \Rightarrow sum = 5
- Observe the relationship between any 2 consecutive windows:
 - Consecutive windows share most elements; except for a change in the first/last element
- Variable-sized sliding window: its size grows and shrinks

Problem #2: Fixed Sliding Window

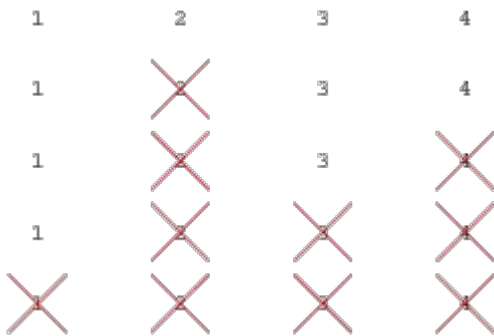
- Read in integers K and N, (where $K \leq N$), and then read $N < 200$ integers.
- Find the **sub-array** (consecutive numbers) of K elements with the maximum sum
- Input 3 7 **1 0 3 -4 2 -6 9**
 - Let's list all sub-arrays of length 3
 - 1 0 3 \Rightarrow sum = 4
 - 0 3 -4 \Rightarrow sum = -1
 - 3 -4 2 \Rightarrow sum = 1
 - -4 2 -6 \Rightarrow sum = -8
 - 2 -6 9 \Rightarrow sum = 5
- Output: 4 6 5 (The sub-array from indices 4 to 6 has a maximum sum of 5)
- Can you do it without nested loops? There are 2 ways

Problem #3: Count Increasing Subarrays

- Read in an Integer N, then read N (< 200) integers.
- Output: **count** how many **subarrays** are **increasing** within the array
- e.g. if the input is 1 2 3 4
 - We can find all sublists of length 1 \Rightarrow [1], [2], [3], [4]
 - All sublists of length 2 \Rightarrow [1, 2], [2, 3], [3, 4]
 - All sublists of length 3 \Rightarrow [1, 2, 3], [2, 3, 4]
 - All sublists of length 4 \Rightarrow [1, 2, 3, 4]
- Example input \Rightarrow output
 - 4 1 2 3 4 \Rightarrow 10 [10 sub-arrays from the previous example, all are increasing]
 - 4 4 3 2 1 \Rightarrow 4 [only sub-arrays of length 1 can be considered]
 - 4 10 20 1 5 \Rightarrow 6
- Easy: 3 nested loops. Medium: using 2 loops. Hard: just 1 loop

Problem #4: Josephus Problem

- Read integers N (< 200) and K (≤ 1000000). *Code for small values of K first*
 - Find the game winner for the following game:
- There is a group of N people in a circle, numbered 1, 2, ..., N
 - We imagine that someone is controlling the game
 - Starting from Person#1, we count K steps - and remove the person at the k th position!
 - We keep doing so until only 1 person remains. The winner!
- Input 4 2
 - We have people at: 1, 2, 3, 4. The controller starts from #1
 - Count 2 people/steps (#2 removed), then start from #3
 - Count 2 people/steps (#4 removed), then start from #1
 - Count 2 people/steps (#3 removed), #1 is the winner
- Output
 - People removed in the order: 2 4 3 1 [same answer as for 10 2. Why?]



Problem #4: Josephus Problem

- Example input \Rightarrow output
 - 7 1 \Rightarrow 1 2 3 4 5 6 7
 - 7 2 \Rightarrow 2 4 6 1 5 3 7
 - 7 3 \Rightarrow 3 6 2 7 5 1 4
 - 7 4 \Rightarrow 4 1 6 5 7 3 2
 - 7 5 \Rightarrow 5 3 2 4 7 1 6
 - 7 6 \Rightarrow 6 5 7 2 1 4 3
 - 7 7 \Rightarrow 7 1 3 6 2 4 5
 - 7 14 \Rightarrow 7 2 6 3 5 4 1
 - 7 1000 \Rightarrow 6 3 2 1 4 7 5
 - 7 99999 \Rightarrow 4 7 5 2 1 3 6

“Acquire knowledge and impart it to the people.”

“Seek knowledge from the Cradle to the Grave.”