

Algorithms and Data Structures

Exercise Session 3



<https://n.ethz.ch/~ahmala/and>

Moodle Quiz Week 4

- 5 minutes
- 5 problems
- Watch out for
 - ☐ Select one or more:
 - ☐ Select one:

Maximum Subarray Problem

- contiguous subset of an array
- Array: [-2, 1, -3, 4, -1, 2, 1, -5, 4]

Password

Password

complexity3

Maximum Subarray Problem

- contiguous subset of an array
- Array: [-2, 1, -3, 4, -1, 2, 1, -5, 4]
- Output: 6 (subarray [4, -1, 2, 1])

Naive Method

Naive Method

- Idea: Check all possible subarrays
- Algorithm
 - Generate all subarrays
 - Calculate sum for each
 - Keep track of maximum sum

Divide and Conquer

Divide and Conquer

- Idea: Recursively divide the array and combine solutions
- Algorithm:
 - Divide array into two halves
 - Recursively find maximum subarray in left and right halves
 - Find maximum subarray crossing the midpoint
 - Find maximum suffix sum of left half
 - Find maximum prefix sum of right half
 - Return the maximum of the three

Master Theorem

Theorem

If $T(n) = aT\left(\left\lceil \frac{n}{b} \right\rceil\right) + O(n^d)$ (for constants $a > 0, b > 1, d \geq 0$), then:

$$T(n) = \begin{cases} O(n^d) & \text{if } d > \log_b a \\ O(n^d \log n) & \text{if } d = \log_b a \\ O(n^{\log_b a}) & \text{if } d < \log_b a \end{cases}$$

Linear Solution

Linear Solution

- Algorithm:
 - Initialize variables for current sum and maximum sum
 - Iterate through the array once
 - At each step, decide whether to start a new subarray or extend the existing one
 - Update maximum sum if current sum is larger

Can we do better than $O(n)$?

Source

<https://cadmo.ethz.ch/education/lectures/HS16/DA/skript/skript.pdf#page=21>

Common Mistake

EXERCISE SHEET 1 AND 2

$$1.1 a) \text{ I.A: } (2 \cdot 0 + 1) = 1 = (0 + 1)^2$$

$$\text{I.H: For any } k \in \mathbb{N}^0: 1 + 3 + 5 + \dots + (2k + 1) = (k + 1)^2$$

$$\text{I.S: } 1 + 3 + 5 + \dots + (2k + 1) + (2k + 3) \stackrel{\text{I.H}}{=} (k + 1)^2 + 2k + 3 = k^2 + 2k + 1 + 2k + 3 = k^2 + 4k + 4 = (k + 2)^2$$

Last Week's Exercise Sheet

Exercise 2.3 *O*-notation mistake. (1 point).

Let $f : \mathbb{N} \rightarrow \mathbb{R}_+$ be a function, with $f(n) \leq O(n)$. A colleague tried to prove that $e^{f(n)} \leq O(e^n)$. You found their notes, in which they start with the statement they want to show, and derive a series of equivalent statements. The notes read:

$$e^{f(n)} \leq O(e^n) \quad \text{— use Definition 1 on the first page} \quad (1)$$

$$e^{f(n)} \leq C \cdot e^n, \text{ for some } C > 0 \quad \text{— take the log on both sides} \quad (2)$$

$$\log(e^{f(n)}) \leq \log(C \cdot e^n) \quad \text{— } \log(C \cdot e^n) = \log C + n \quad (3)$$

$$f(n) \leq \log C + n \quad \text{— } n + \log C \leq O(n) \quad (4)$$

$$f(n) \leq O(n) \quad \text{— True by assumption, so we are done!} \quad (5)$$

Forward & Backward Proofs

Omega Notation

Definition 1 (Ω -Notation). For $f : N \rightarrow \mathbb{R}_+$,

$$\Omega(f) := \{g : N \rightarrow \mathbb{R}_+ \mid f \leq O(g)\}.$$

We write $g \geq \Omega(f)$ instead of $g \in \Omega(f)$.

Theta Notation

Definition 2 (Θ -Notation). For $f : N \rightarrow \mathbb{R}_+$,

$$\Theta(f) := \{g : N \rightarrow \mathbb{R}_+ \mid g \leq O(f) \text{ and } f \leq O(g)\}.$$

We write $g = \Theta(f)$ instead of $g \in \Theta(f)$.

In other words, for two functions $f, g : N \rightarrow \mathbb{R}_+$ we have

$$g \geq \Omega(f) \Leftrightarrow f \leq O(g)$$

and

$$g = \Theta(f) \Leftrightarrow g \leq O(f) \text{ and } f \leq O(g).$$

Code Expert

First programming task on Friday

40% of the exam

You get one point for each passing test set. To pass both test sets correctly, your solution has to be in $O(\log n)$ time.



Theory will only take you so far

Exercises

Filters: Show more ▾ Sort by date ▴▾

🔴 To do 🟠 Incomplete 🟢 Complete 📄 Feedback

Warm-Up Assignment Closed 2 years ago
🔗 Solutions available

- 🟢 Welcome
- 🟢 Median of Three

Programming Assignment 1 4 XP Closed 2 years ago
🔗 Solutions available

- 🟢 1.1 Nearest Multiple 2 XP
- 🟢 1.2 Longest Alternating Subarray 2 XP

Programming Assignment 2 4 XP Closed 2 years ago
🔗 Solutions available

- 🟢 2.1 Range Counting 2 XP
- 🟢 2.2 Triangles 2 XP

Median of Three - Student attempt

```
1- class Main {
2-
3-     public static void main(String[] args) {
4-         int round = In.readInt(); // read the number of actions
5-         for (int i = 0; i < round; i++) {
6-             Out.println(median(In.readInt(), In.readInt(), In.readInt()));
7-         }
8-     }
9-
10-     public static int median(int a, int b, int c) {
11-         // TODO: complete this method
12-         return 0;
13-     }
14- }
```

Ahmet Ala

Not submitted yet
Your best submission counts

Create new submission

Median of Three

Your task is to write a program that, given three distinct integers, returns their median. For example, if the input is 3, 1 and 2, the output is 2. You only need to implement the method "median" in the file "Main.java".

Peer Grading

Task 2.3