

*Introduction to Algorithms: 6.006*

Massachusetts Institute of Technology

Instructors: Erik Demaine, Jason Ku, and Justin Solomon

Problem Set 3

---

## Problem Set 3

---

**Name:** Heejin Chae

**Collaborators:** Name1, Name2

---

**Problem 3-1.**

(a)

(b)

**Problem 3-2.****(a)****(b)****(c)**

**Problem 3-3.**

- (a)
- (b)
- (c)
- (d)

**Problem 3-4.****(a)****(b)**

**Problem 3-5.**

- (a) idea : Turn string into base 26 integer sequence. For example "abcdde" becomes "11121...0". Each position corresponding alphabet place.

First prove build time is  $O(|A|)$

Building a hash table from A takes  $O(k) + n * O(1)$ . For first k chunks, the initial frequency is established by iterating over k elements in A. Then by sliding 1 element side by side it doesn't need to be rebuilt. But by memorizing previous first item, decrement the corresponding frequency and increment the one for the next added one. So by repeating this process, The overall time complexity is  $O(k) + n * O(1)$  which  $O(k)$  is thrown by first build and  $n * O(1)$  for sliding windows

(b)

- (c) Submit your implementation to `alg.mit.edu`.