

# Problem Set 1

Algoritmi e Strutture Dati – a.a. 2025/2026

Università di Roma “Tor Vergata”

Prof. Luciano Gualà

## Importanti avvertenze preliminari:

1. Il gruppo deve essere formato da almeno tre persone!
2. Per lo svolgimento si consiglia fortemente l'utilizzo del template L<sup>A</sup>T<sub>E</sub>X disponibile nella pagina web del corso.
3. Lettere il documento *Problem Set: Istruzioni e consigli* sulla pagina web del corso.

**Linee guida per la consegna.** Gli elaborati dovranno essere consegnati entro mercoledì 10/12/2025 alle ore 23:59. Si prega di seguire le seguenti indicazioni:

1. le soluzioni dovranno essere consegnate via email a entrambi gli indirizzi [gual@mat.uniroma2.it](mailto:gual@mat.uniroma2.it) e [alessandrostr95@gmail.com](mailto:alessandrostr95@gmail.com);
2. l'oggetto dell'email dovrà essere “Consegna problem set 1 - 2025/2026”;
3. il file dovrà essere in formato pdf;
4. il nome del file dovrà essere “PS\_1\_XXX.pdf” dove XXX è il primo cognome (in lower case) in ordine alfabetico dei membri del gruppo;
5. l'email dovrà essere inviata tramite l'indirizzo dello studente XXX presente nel nome del file;
6. inserire nel file nome, cognome, indirizzo email e matricola di tutti i membri del gruppo (meglio ancora se tutti i membri del gruppo sono in indirizzo nell'email della consegna).

Buon lavoro 🔥 😎!

**Problema 1** (*Aspettando natale*)

Natale sta arrivando e la tradizione vuole che prepariate l'albero, il presepe, le luci in balcone e tutto il resto. Avete messo il materiale natalizio nella cassetiera della cantina che ha  $n$  cassetti numerati da 1 a  $n$ . Siete sicuri che la roba occupa  $\ell$  cassetti consecutivi ma non ricordate né quali sono, né quanti sono (cioè non ricordate quanto vale  $\ell$ ). L'unico modo per scoprirlo è aprire i cassetti. Ma in che ordine? Da quando seguite il corso di algoritmi vi è presa la mania di ottimizzare ogni cosa e quindi ora vi chiedete: che strategia mi conviene utilizzare?

Progettate un algoritmo che, pur non conoscendo  $\ell$ , è in grado di trovare un cassetto contenente materiale natalizio apprendone nel caso peggiore  $O(n/\ell)$ . C'è un algoritmo migliore? Argomentate sull'eventuale ottimalità asintotica dell'algoritmo proposto.

**Problema 2** (*Straziota alla materna*)

Il dottor maestro Straziota ha organizzato una competizione di informatica alla scuola materna del suo quartiere e ora che la gara si è conclusa deve affrontare il momento della premiazione cosa che, per ragioni che capirete a breve, non è affatto facile. Ha fatto mettere i giovani studenti uno accanto all'altro a formare una linea. Ogni studente nella competizione ha preso un certo punteggio numerico. In particolare, l' $i$ -esimo bambino della linea ha preso  $x_i$ . L'intenzione di Straziota è di premiare i partecipanti con delle caramelle<sup>1</sup>, ma deve seguire rigorosamente due principi per non scontentare (i genitori di) alcun bambino:

- Tutti devono vincere qualcosa, ovvero ogni bambino deve ricevere almeno una caramella;
- Due bambini che si trovano uno accanto all'altro devono ricevere un numero di caramelle che riflette l'ordine relativo dei punteggi: il bambino che ha ricevuto il punteggio più alto deve ricevere più caramelle del compagno accanto a lui (mentre punteggi uguali devono implicare lo stesso numero di caramelle).<sup>2</sup> Infatti, se così non fosse uno dei due bambini – quello che ha preso meno caramelle – si lamenterebbe con i propri genitori dell'ingiustizia.

Ora, queste caramelle, sono decisamente costose così Straziota, da buon algoritmista, vuole usarne il meno possibile, senza venire meno ai due principi. Aiutate Straziota progettando un algoritmo efficiente che calcola quante caramelle assegnare a ciascun bambino. Discutete la correttezza e la complessità temporale della soluzione proposta.

---

<sup>1</sup>Le caramelle sono di un tipo specifico approvato esplicitamente dal consiglio dei genitori: sono senza lattosio, senza glutine, senza zuccheri saturi, senza spartame, senza olio di palma e senza sostanze psicotrope.

<sup>2</sup>Alcuni genitori hanno sponsorizzato questa regola sbandierando la necessità di un sistema meritocratico. Un algoritmista attento si renderà conto che questa regola non garantisce da sola una distribuzione meritocratica delle caramelle fra tutti i bambini. Riuscite a vedere perché?

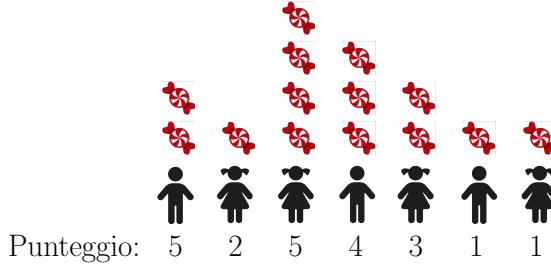


Figure 1: Esempio di istanza per il Problema 2. Sotto i bambini sono riportati i rispettivi punteggi, mentre sopra le caramelle che ricevono (rispettando i due principi). In questo esempio il minor numero di caramelle è 14.

### Problema 3 (*Una struttura dati per prefissi*)

Dovete progettare una struttura dati che mantenga una sequenza di elementi su cui è possibile chiedere informazioni relative ai prefissi della sequenza. Formalmente le operazioni che la struttura dati deve supportare sono le seguenti:

- **build( $V$ ):** prende in input un vettore  $V[1 : n]$  di  $n$  numeri e costruisce la struttura dati relativa alla sequenza descritta da  $V$ .
- **min( $i$ ):** prende un indice  $i \in \{1, \dots, n\}$  e restituisce il valore minimo dei primi  $i$  elementi della sequenza.
- **sum( $i$ ):** prende un indice  $i \in \{1, \dots, n\}$  e restituisce la somma dei primi  $i$  elementi della sequenza.
- **update( $i, x$ ):** aggiorna il valore dell' $i$ -esimo elemento della sequenza assegnandogli il valore  $x$ .

L'operazione di **build** deve avere complessità temporale  $O(n)$  mentre tutte le altre devono costare  $O(\log n)$ . La memoria usata dalla struttura dati deve essere lineare nella lunghezza della sequenza.