



Lecturers: **Evangelos Kipouridis and Tomasz Kociumaka**
Tutors: **Anouk Duyster and Yonggang Jiang**

Winter 2025/26

————— **Randomized and Approximation Algorithms, Exercise Sheet 6** —————

<https://www.mpi-inf.mpg.de/departments/algorithms-complexity/teaching/winter-2025/26/randomized-and-approximation-algorithms>

Total Points: **100** + 50 bonus points

Due: **14:00** on Thursday, **December 4**, 2025

————— **Exercise 1** —————

20 points

Design an $\mathcal{O}(|V| + |E|)$ -time deterministic algorithm that, given a *weighted directed* graph $G = (V, E, w)$ without self-loops, constructs a cut $(S, V \setminus S)$ of cost at least $\frac{1}{4}w(E)$, where $w(E) = \sum_{e \in E} w(e)$.

Provide all necessary implementation details (in pseudocode or text).

Hint: Recall Exercise 1 from Sheet 2.

————— **Exercise 2** —————

10 + 10 points + 20 bonus points

Recall the deterministic algorithm that constructs an independent set of size at least $\sum_{v \in V} \frac{1}{\deg(v)+1}$. The implementation from the lecture uses real-valued variables ϕ_v , which we now wish to avoid.

Let $U = V \setminus \bigcup_{s \in S} N[s]$ denote the set of *candidates* (vertices that can still be added to the independent set S). For each vertex v , define the *penalty*

$$\phi_v = \sum_{u \in N[v] \cap U} \frac{1}{1 + \deg(u)}.$$

A candidate v is called *good* if $\phi_v \leq 1$.

- a Prove that if $U \neq \emptyset$, then the candidate $v \in U$ minimizing $\deg(v)$ is good.
- b Design an $\mathcal{O}(|V| + |E|)$ -time deterministic algorithm that constructs an independent set of size at least $\sum_v \frac{1}{\deg(v)+1}$, using only pointers and $\mathcal{O}(\log |V|)$ -bit integer variables.
- c (*) Provide an analogous analysis and $\mathcal{O}(|V| + |E|)$ -time implementation of a deterministic algorithm that repeatedly chooses a candidate $v \in U$ minimizing $|N[v] \cap U|$ (instead of $|N[v]| = \deg(v) + 1$).
Hint: Ensure that no logarithmic overhead arises from priority queues.

————— **Exercise 3** —————

5 + 15 + 5 + 15 points

In this exercise, we consider a multiplayer casino setup similar to Exercise 4 from Sheet 4.

There are m players and n rounds. For $i \in [1..m]$ and $j \in [0..n]$, let $b_{i,j}$ denote the budget of player i after j rounds. The initial budget of player i is $b_{i,0} := 1$, and their final score is $b_{i,n}$.

Each round j proceeds as follows:

- Each player $i \in [1..m]$ chooses a guess $g_{i,j} \in \{0, 1\}$ and a wager $w_{i,j} \in [0, b_{i,j-1}]$.
- The casino chooses $x_j \in \{0, 1\}$ (uniformly at random if it is a fair casino).
- Each player updates their budget as

$$b_{i,j} := \begin{cases} b_{i,j-1} + w_{i,j} & \text{if } g_{i,j} = x_j, \\ b_{i,j-1} - w_{i,j} & \text{otherwise.} \end{cases}$$

For a fair casino, the expected final score of each player satisfies $\mathbb{E}[b_{i,n}] = b_{i,0} = 1$, independently of the player's strategy.

- a** Prove that there exists an (unfair, adversarial) strategy for the casino that guarantees $\sum_i b_{i,n} \leq m$, independently of the players' strategies. Provide a linear-time implementation of the casino's strategy assuming that, in each round, you are given a list of nonzero wagers and the corresponding guesses.
Hint: In all parts of this exercise, you may assume constant-time arithmetic operations on real numbers.

Now, recall the *set discrepancy problem*. We interpret the casino as deciding, for subsequent elements u of the universe $U = [1..|U|]$, whether to assign u to V or W . The discrepancy of a set $S \subseteq U$ is defined as

$$D(S) := |V \cap S| - |W \cap S|.$$

- b** For a given set $S \subseteq U$ and threshold $t \in \mathbb{Z}_{\geq 0}$, describe a player strategy that guarantees a final score of at least $\exp(\frac{t^2}{2|S|})$ whenever $D(S) \geq t$, and symmetrically a strategy guaranteeing the same bound when $-D(S) \geq t$. The strategies should be valid against any adversarial casino.
Hint: Use Exercise 4d from Sheet 4, where the underlying strategy is to bet a δ fraction of the current budget. Do not forget about the special case $t > |S|$, which corresponds to $\delta > 1$.
- c** For a sequence of sets $S_1, \dots, S_s \subseteq U$, design a joint strategy for $2s$ players that guarantees a total final score strictly greater than $2s$ whenever $|D(S_i)| \geq \sqrt{2|S_i| \ln(2s)}$ for some $i \in [1..s]$.
Hint: Note that $\sqrt{2|S_i| \ln(2s)}$ is never an integer (nor an algebraic number).
- d** Combine all parts above to design an $\mathcal{O}(|U| + \sum_i |S_i|)$ -time deterministic algorithm that, given sets S_1, \dots, S_s , partitions U into V and W so that

$$|D(S_i)| \leq \sqrt{2|S_i| \ln(2s)} \quad \text{for all } i \in [1..s].$$

Hint: You may assume constant-time arithmetic operations on real numbers, including $\sqrt{\cdot}$, $\ln(\cdot)$, and $\lceil \cdot \rceil$.

— **Exercise 4** ————— **5 + 15 points + 15 + 15 bonus points** —

Consider a sequence $T[0..n-1]$ of n elements from a universe $U := [0..n-1]$ and a window size $w \in [1..n]$ such that, for every $i \in [0..n-w]$, the elements $T[i], T[i+1], \dots, T[i+w-1]$ are distinct.

For an order \prec on U , define

$$S_{\prec} := \{i \in [0..n-w] : T[i] = \min_{\prec} \{T[i], \dots, T[i+w-1]\}\} \cup \{i \in [w-1..n-1] : T[i] = \min_{\prec} \{T[i-w+1], \dots, T[i]\}\}.$$

- a** Prove that $\mathbb{E}[|S_{\prec}|] \leq \frac{2(n-w+1)}{w}$ when the order \prec is chosen uniformly at random.
- b** Design a deterministic $n^{\mathcal{O}(1)}$ -time algorithm that finds an order \prec such that $|S_{\prec}| \leq \frac{2(n-w+1)}{w}$.
- c** (*) Design a deterministic $\mathcal{O}(n)$ -time algorithm that finds an order \prec such that $|S_{\prec}| \leq \mathcal{O}(\frac{n-w+1}{w})$.
- d** (*) Design a deterministic $\mathcal{O}(n)$ -time algorithm that finds an order \prec such that $|S_{\prec}| \leq \frac{2(n-w+1)}{w}$.