Final Exam, Computational Complexity 2024

- Books, notes, cellphones, etc. are **not** allowed.
- Your explanations and proofs should be clear enough and in sufficient detail so that they are easy to understand and have no ambiguities.
- You are allowed to freely refer to material covered in lectures/exercises.
- **Do not touch until the start of the exam.**

Good luck!

Solution sketch

Name: ______________________ **N° Sciper:** ______________

1 *(15 pts)* **Quick-fire round.** Consider the following statements.

1. $\mathbf{BPP} \subseteq \mathbf{NP}$
2. $\mathbf{L} \neq \mathbf{PSPACE}$
3. $\mathbf{PSPACE} = \mathbf{NPSPACE}$
4. If $\mathbf{P} = \mathbf{NP}$, then $\mathbf{PH} = \mathbf{P}$.
5. If $\mathbf{PH} = \mathbf{PSPACE}$, then $\mathbf{PH}$ collapses.
6. If $A \in \mathbf{P}^B$, then $A \leq_p B$.
7. If $\mathbf{NP} = \mathbf{coNP}$, then $\mathbf{NP}^A = \mathbf{coNP}^A$ for all oracles A.
8. Savitch's algorithm solves *st*-connectivity in polynomial time.
9. Language $\{\langle G\rangle : G$ is a graph with vertex cover of size 2024$\}$ is **NP**-complete.
10. Language $\{\langle \varphi\rangle : \varphi$ is a CNF that has at least 2024 satisfying assignments$\}$ is **NP**-complete.
11. Language $\{\langle \varphi\rangle : \varphi$ is a satisfiable CNF whose clauses have *two* literals$\}$ is **NP**-complete.
12. Language $\{\langle C\rangle : C$ is a boolean circuit such that $C(x) = 1$ for all $x\}$ is **NP**-complete.
13. Every CNF formula admits a Resolution refutation.
14. Every CNF formula φ can be equivalently written as a DNF formula ψ, that is, $\varphi(x) = \psi(x)$ for all x.
15. There is a *monotone* boolean function f whose decision tree depth complexity is larger than its certificate complexity, that is, $\mathrm{D}(f) > \mathrm{C}(f)$.

For each statement, write down one of

- **T** if the statement is known to be true.
- **F** if the statement is false *or not known to be true.* E.g., both $\mathbf{P} = \mathbf{NP}$ and $\mathbf{P} \neq \mathbf{NP}$ should be marked **F**.
- or leave your answer empty.

A correct **T**/**F** answer is worth +1 point, an incorrect answer is worth −1 point, and an empty answer is worth 0 points.

1.	F
2.	T
3.	T
4.	T
5.	T

6.	F
7.	F
8.	F
9.	F
10.	T

11.	F
12.	F
13.	F
14.	T
15.	T

2 *(15 pts)* **Problem Garden.** For each problem below, classify it into the smallest possible complexity class (as seen in the course). You do not need to justify your answers (but you can!).

1. **Input:** Graph G and integer k. **Output:** YES iff the largest clique in G is of size k.
2. **Input:** A DNF formula φ and a unary integer 1^k. **Output:** YES iff there is a CNF formula with at most k clauses that is equivalent to φ.
3. **Input:** Bipartite graph G. **Output:** YES iff G contains a perfect matching.
4. **Input:** Boolean circuit C with $2n$ input bits. **Output:** YES iff there is a directed path from 0^n to 1^n in the directed graph $G = (\{0,1\}^n, E)$ defined by $(u,v) \in E$ iff $C(u,v) = 1$.
5. **Input:** Undirected graph G. **Output:** YES iff G contains a cycle of odd length.

1) DP 2) Σ_2^p 3) P 4) PSPACE 5) L

3 *(15 pts)* **NP-completeness.** Suppose $\mathcal{S} = \{S_1, \dots, S_n\}$ is a family of subsets of $\{1, 2, \dots, n\}$. We say that k distinct sets $S_{i_1}, \dots, S_{i_k} \in \mathcal{S}$ form a *packing of size* k if the sets are pairwise disjoint, that is, $S_{i_j} \cap S_{i_{j'}} = \emptyset$ for $j \neq j'$. Show that the following problem is **NP**-complete:

$$\text{SetPacking} = \{ \langle \mathcal{S}, k \rangle : \text{Family } \mathcal{S} \text{ contains a packing of size } k \}.$$

SET PACKING $\in NP$

just check the conditions

INDEPENDENTSET $\leq$ SET PACKING

$(G, k) \leq (S, k)$

where $S = \{S_v\}_{v \in V(G)}$ is a collection of subsets of $E(G)$ ← has size $\leq n^2$
constructed as follows.

- $S_v = \{e \in E(G) : v \in e\}$
- Now, just pad S with $|E(G)| - |V(G)|$ copies of $E(G)$ to respect size constraints
- Observe $S_{v_1} \cap S_{v_2} \neq \emptyset \iff \{v_1, v_2\} \in E(G)$

4 *(15 pts)* **Oracles.** A *k-query oracle Turing machine* is an oracle Turing machine that is permitted to make at most k queries on each input. Define $\mathbf{P}^{A,k}$ to be the collection of languages that are decidable by polynomial-time k-query oracle Turing machines with an oracle for A. Note that

$$\mathbf{NP} \cup \mathbf{coNP} \subseteq \mathbf{P}^{\mathrm{SAT},1}.$$

Show that, if $\mathbf{NP} \neq \mathbf{coNP}$, then the above inclusion is strict, that is, $\mathbf{NP} \cup \mathbf{coNP} \neq \mathbf{P}^{\mathrm{SAT},1}$.

if $NP \neq coNP$ then $SAT \notin coNP$, $UNSAT \notin NP$ as they are complete

consider $\mathcal{L} = \{ \langle \varphi, b \rangle : \varphi \in SAT$ and $b = 0$; $\varphi \in UNSAT$ and $b = 1 \}$

we have $SAT \leq \mathcal{L}$ by setting $b=0$
$UNSAT \leq \mathcal{L}$ by setting $b=1$

thus $\mathcal{L} \notin NP \cup coNP$ (as $NP \neq coNP$)

on the other hand, $\mathcal{L} \in P^{NP,1}$:

if $b=0$, return result of ORACLE on φ
if $b=1$, return negation of ORACLE on φ

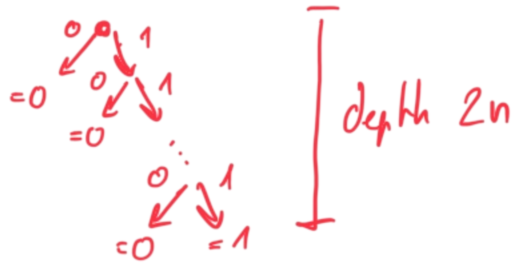
5 *(20 pts)* **Decision trees.** Let $f\colon \{0,1\}^n \to \{0,1\}$ be a boolean function and let $\textsc{Xor}_2\colon \{0,1\}^2 \to \{0,1\}$ denote the 2-bit parity function ($\textsc{Xor}_2(x_1,x_2) = 1$ iff x_1+x_2 is odd). Define the composed function $f\circ\textsc{Xor}_2$ by $(f\circ\textsc{Xor}_2)(x,y) := f(\textsc{Xor}_2(x_1,y_1),\dots,\textsc{Xor}_2(x_n,y_n))$ where $x,y\in\{0,1\}^n$.

5a Show that for every f, the function $f\circ\textsc{Xor}_2$ requires decision trees of *size* at least $2^{\mathrm{D}(f)}$ (where, as usual, $\mathrm{D}(f)$ denotes the decision tree *depth* complexity of f).

5b Show that **5a** fails if we use the $\textsc{And}_2$ function in place of $\textsc{Xor}_2$. That is, exhibit some n-bit function f such that $f\circ\textsc{And}_2$ admits a decision tree of size less than $2^{\mathrm{D}(f)}$.

a) use Tree-Adversary game. whenever tree ask for a bit in group (x_i, y_i) for the first time, can let Tree choose value (and win a point) if it is the second time, set it accordingly to best adversary for f -> can let Tree choose at least $D(f)$ queries before fixing the value, hence $D\text{size}(f\circ \text{XOR}_2) \geq 2^{D(f)}$

b) let $f := \text{AND}_n$, then $D\text{size}(\text{AND}_n \circ \text{AND}_2) = D\text{size}(\text{AND}_{2n}) \leq O(n)$ using unbalanced tree:



6 *(20 pts)* **Communication complexity.** Let $F\colon \mathcal{X}\times\mathcal{Y}\to\{0,1\}$ be a two-party communication problem where Alice gets $x\in\mathcal{X}$ and Bob gets $y\in\mathcal{Y}$. A set $S\subseteq\mathcal{X}\times\mathcal{Y}$ is called a 0-*fooling set* iff (i) $F(x,y)=0$ for all $(x,y)\in S$, and (ii) for any two distinct $(x,y),(x',y')\in S$ we have $F(x,y')=1$ or $F(x',y)=1$. Let $\mathrm{fs}_0(F)$ denote the largest size $|S|$ of a 0-fooling set S for F.

6a Show that $\mathrm{N}_0(F)\geq\log\mathrm{fs}_0(F)$ for every F. (Here $\mathrm{N}_0(F)=\mathrm{N}_1(\neg F)$ is the co-nondeterministic communication complexity.)

6b Show that $\mathrm{fs}_0(\mathrm{SI}_n)=2^n$ where $\mathrm{SI}_n(x,y):=\bigvee_{i=1}^n(x_i\wedge y_i)$ is the set-intersection problem.

(a) 2^N_0(F) >= C_0(F) := 0-covering number of F.
Let S be a maximal fooling set. For two different (x,y), (x’,y’) in S, if they are in the same rectangle R, then (x,y’) and (x’,y) are both in R. By the definition of fooling set, R is not monochromatic. Thus (x,y) and (x’,y’) must be in two different monochromatic 0-rectangles.
So 2^N_0(F) >= C_0(F) >= |S| = fs_0(F).

(b) UB: fs_0(SI_n) <= 2^N_0(F) <= 2^n
LB: S = {(x,\neg x) : x \in {0,1}^n}, |S|=2^n.
Verify S is a fooling set:
(i) For all x \in {0,1}^n, SI(x,\neg x) = 0.
(ii) For x != y, exist some i, x_i != y_i. Either x_i = \neg y_i=1, or \neg x_i = y_i=1. Thus SI(x,\neg y)=1 or SI(\neg x,y)=1.