CS-524: Computational Complexity
Prof. Mika Göös
17 December 2021

1

Sample Exam

SCIPER : **222222**

Signature :

Final Exam, Computational Complexity 2021

- Books, notes, communication, calculators, cell phones, computers, 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 refer to material covered in lectures and exercises including theorems without reproving them.
- Please answer inside the space indicated (the exams will be scanned for grading purposes).
- **Do not touch until the start of the exam.**

Good luck and have fun!

Problem 1 / 15 points	Problem 2 / 25 points	Problem 3 / 20 points	Problem 4 / 20 points	Problem 5 / 20 points

Total / 100

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

1. 3-SAT $\leq_p$ 2-SAT.
2. $\mathsf{TIME}(n) \neq \mathsf{TIME}(n^{2021})$.
3. If $A \leq_p B$, then $A \in \mathsf{P}^B$.
4. If $\mathsf{P} = \mathsf{NP}$, then $\mathsf{P}^A = \mathsf{NP}^A$ for every oracle A.
5. There exists an oracle A such that $\mathsf{P}^A = \mathsf{EXP}^A$.
6. Directed s–t connectivity is in L.
7. PH has a complete problem.
8. If $\mathsf{NP} \subseteq \mathsf{P/poly}$, then PH collapses to its first level, i.e., $\mathsf{PH} = \mathsf{NP} = \mathsf{coNP}$.
9. $\mathsf{NSPACE}(n) \subseteq \mathsf{SPACE}(n^2)$.
10. If $\mathsf{NP} \subseteq \mathsf{BPP}$, then $\mathsf{NP} \subseteq \mathsf{RP}$.
11. Graph isomorphism is in coAM.
12. Every n-bit function computed by a $\mathsf{poly}(n)$-sized circuit can also be computed with a $\mathsf{poly}(n)$-sized CNF.
13. Every n-bit function can be computed by a depth-$O(n)$ circuit with fan-in 2 gates.
14. If a boolean function can be written both as k-DNF and k-CNF, then it admits a depth-$O(k)$ decision tree.
15. Certificate complexity $\mathrm{C}(f)$ and sensitivity $\mathrm{s}(f)$ satisfy $\mathrm{C}(f) \leq \mathrm{s}(f)$ for all functions f.

For each box below, write one of the following symbols:

- **T** if the statement is known to be true.
- **F** if the statement is false *or not known to be true* (e.g., both $\mathsf{P} = \mathsf{NP}$ and $\mathsf{P} \neq \mathsf{NP}$ should be marked **F**).
- or leave the box 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.

Your answers:

1.		6.		11.	
2.		7.		12.	
3.		8.		13.	
4.		9.		14.	
5.		10.		15.	

2. Complexity Zoo. *(25 pts)* Consider the following list of complexity classes:

$$\mathsf{P},\quad \mathsf{NP},\quad \mathsf{coNP},\quad \mathsf{EXP},\quad \mathsf{PSPACE},\quad \mathsf{L},\quad \mathsf{PH},\quad \Sigma_2\mathsf{P},\quad \Pi_2\mathsf{P},\quad \mathsf{P}^{\mathsf{NP}},\quad \mathsf{BPP},\quad \mathsf{IP},\quad \mathsf{P/poly}.$$

Draw a ***class inclusion diagram*** for the above classes that has a directed arrow $A \to B$ between classes A and B whenever $A \subseteq B$. You do not have to draw *transitive arrows*, i.e., if $A \to B$ and $B \to C$, you can leave out $A \to C$. For ease of readability, please draw the smallest class at the bottom, and the largest class at the top of the page.

Please answer inside the box

3. Problem Garden. *(15 pts)*

(a) For each problem below, classify it into the smallest possible complexity class (as seen in lectures/exercises). If you use a class not mentioned in Problem 2, then relate your class to those in Problem 2 by inclusions. (You do not need to justify your answers.)

1. **Input:** DNF formula φ. **Output:** YES iff $\varphi(x) = 1$ for some x.
2. **Input:** DNF formula φ. **Output:** YES iff $\varphi(x) = 1$ for all x.
3. **Input:** Graph G and number k. **Output:** YES iff the smallest vertex cover in G is of size exactly k.
4. **Input:** Boolean circuit C. **Output:** YES iff C is the smallest circuit that computes the same function as C.
5. **Input:** Arithmetic circuit C over integers $\mathbb{Z}$. **Output:** YES iff $C(x) = 0$ for all $x \in \mathbb{Z}^n$.

Please answer inside the box

(b) Give an example (including a precise definition) of a PSPACE-complete problem.

4. NP-completeness. *(20 pts)* Let $G = (V, E)$ be an undirected graph. A set of vertices $D \subseteq V$ is *dominating* if every vertex $v \in V \setminus D$ has a neighbour in D (that is, $\{v, u\} \in E$ for some $u \in D$). Define

$$\textsc{DominatingSet} = \{\langle G, k\rangle : G \text{ has a dominating set of size at most } k\}.$$

Prove that DOMINATINGSET is NP-complete. In your proof, you may assume the NP-completeness of any of the problems discussed in class (SAT, Independent Set, Clique, Subset Sum, Vertex Cover, Set Cover, etc.).

Please answer inside the box

5. Communication complexity. *(20 pts)*

(a) In the $\textsc{Index}_n$ problem, Alice holds $x \in \{0,1\}^n$, Bob holds $i \in [n]$, and their goal is to output $\textsc{Index}(x,i) = x_i \in \{0,1\}$ (i-th bit of x). Show that any deterministic *one-way communication protocol* (Alice sends a single message to Bob who then outputs the value of the function) for $\textsc{Index}_n$ requires $\Omega(n)$ bits of communication.

Please answer inside the box

(b) Define the *Set-Intersection* problem and then use it to show that the following two-party communication problem requires $\Omega(n)$ bits of communication (even for randomised protocols):

- Alice holds a graph $G_A = ([n], E_A)$;
- Bob holds a graph $G_B = ([n], E_B)$;
- Decide whether the union graph $G_A \cup G_B = ([n], E_A \cup E_B)$ contains a perfect matching.

Please answer inside the box