

CS570 Summer 2025: Analysis of Algorithms**Exam III**

	Points		Points
Problem 1	18	Problem 4	18
Problem 2	10	Problem 5	16
Problem 3	18	Problem 6	18
Total 98			

First name	
Last Name	
Student ID	

Instructions:

1. This is a 2-hr exam. Closed book and notes. No electronic devices or internet access.
2. A single double sided 8.5in x 11in cheat sheet is allowed.
3. If a description to an algorithm or a proof is required, please limit your description or proof to within 150 words, preferably not exceeding the space allotted for that question.
4. No space other than the pages in the exam booklet will be scanned for grading.
5. If you require an additional page for a question, you can use the extra page provided within this booklet. However please indicate clearly that you are continuing the solution on the additional page.
6. Do not detach any sheets from the booklet.
7. If using a pencil to write the answers, make sure you apply enough pressure, so your answers are readable in the scanned copy of your exam.
8. Do not write your answers in cursive scripts.
9. This exam is printed double sided. Check and use the back of each page.

1) 18 pts (2 pts each)

Mark the following statements as **TRUE**, **FALSE** or **UNKNOWN** by circling the correct answer. No need to provide any justification.

[**TRUE**/FALSE/**UNKNOWN**]

If a problem is in NP, it must be solvable in polynomial time.

[**TRUE**/**FALSE**/UNKNOWN]

If $P \neq NP$, then no NP problem can be solved in polynomial time.

[**TRUE**/FALSE/UNKNOWN]

A ρ -approximation algorithm for the general TSP for any constant ρ would imply that $P = NP$.

[**TRUE**/FALSE/UNKNOWN]

Let Y be the problem of finding out if there is a feasible circulation in a given circulation network. If a problem X is NP-complete and $X \leq_p Y$, then $P = NP$.

[**TRUE**/**FALSE**/UNKNOWN]

Every NP-hard decision problem belongs to NP.

[**TRUE**/ FALSE/UNKNOWN]

If we find a polynomial time solution to 3SAT, it implies that we have also found polynomial time solutions to all NP-Intermediate problems.

[**TRUE**/**FALSE**/UNKNOWN]

If $P=NP$ this means that there is a polynomial time solution for the Halting problem.

[**TRUE**/FALSE]

The feasible region of a linear program is always a convex polytope.

[**TRUE**/**FALSE**]

Given that there is a 2-approximation algorithm for Vertex Cover, we know that there is a polynomial-time 1/2-approximation algorithm for the Independent Set problem.

2) 10 pts **Select all correct answers!** No partial credit

I. Given a directed unweighted graph G , two of its vertices S and T , and a number k , which of the following decision problems are NP-complete? (3 pts)

- a) Is there a simple path of length at most k from S to T ?
- b) **Is there a simple path of length at least k from S to T ?**
- c) Are there at least k simple edge-disjoint paths from S to T ?
- d) Are there at most k simple edge-disjoint paths from S to T ?

II. If graph G with 20 vertices has a Vertex Cover of size 5, then which of the following is a possible size for the **maximum** Independent Set of graph G ? (3 pts)

- a) 5
- b) 10
- c) **15**
- d) **20**
- e) 25

III. Which of the following problems are known to be NP-complete? (4 points)

- a) 3-SAT
- b) Integer Linear Programming (with binary variables)
- c) Shortest Path in a Graph with Positive Weights
- d) Subset Sum
- e) Hamiltonian Cycle in Undirected Graphs
- f) Topological Sorting of a Directed Acyclic Graph (DAG)
- g) Maximum Flow in a Flow Network
- h) Bipartite Matching in a Graph

Correct Answers:

A. 3-SAT

B. Integer Linear Programming (0-1 ILP)

D. Subset Sum

E. Hamiltonian Cycle in Undirected Graphs

3) 18 pts

Suppose you are selecting a team of athletes from N available athletes. There are M different skills that determine the profile of any athlete. For each skill i , a subset G_i of athletes are really good at it, and a subset B_i of athletes are really bad at it (G_i and B_i do not overlap, but there may be athletes that are neither in G_i nor B_i , i.e., they are neither really good, nor really bad in that skill). The team you select will be considered **“weak” w.r.t. skill i** if

- 1) No athlete from G_i (really good) is included in the team, **AND**
- 2) ALL athletes in B_i (really bad) are included in the team

You want to form a team of size at least k which is not weak w.r.t. any of the skills. Prove that the Athlete Team Selection (ATS) problem is

- a) in NP. (6 pts)
- b) NP-Hard. (Hint: Consider using the general satisfiability problem (SAT) for your reduction) (12 pts)

Solution:

For containment in NP: A set of athletes as a certificate. Certifier checks that there are at least k of them, and evaluates the two conditions for each of the skills - which takes $O(N)$ time for each of the M skills - and check that the team is not weak w.r.t. any of the skills.

For NP-Hardness, reduce from SAT:

Construct an athlete a_i for each variable x_i , and skill s_j for each clause c_j .

For each skill s_j , $G_j = \{a_i \mid \text{var } x_i \text{ appears as a positive literal in clause } c_j\}$ and similarly, $B_j = \{a_i \mid \text{var } x_i \text{ appears as a negated literal in clause } c_j\}$. Set $k=0$.

Claim: The SAT instance has a satisfying assignment if and only if the constructed ATS instance has a solution.

Proof:

\Rightarrow) Suppose there is a satisfying assignment for the SAT instance. Given the assignment, we select athletes corresponding to the True variables. Now, (to show contradiction), suppose this team was weak for a skill s_j , none of the athletes in G_j are selected and all in B_j are selected. This means none of the positive literals in clause c_j are true, and all the variables that are negated are true, which makes c_j not satisfied, a contradiction (since we had a satisfying assignment). This means the team described above is not weak w.r.t. any skill.

Student ID:

\Leftarrow) Suppose there is a team of athletes that is not weak w.r.t. any skill. For every selected athlete a_i , we set x_i to true and for any a_i not on the team, we set it false. Now, (to show contradiction), suppose this assignment makes c_j NOT satisfied, this means none of the positive literals in clause c_j are true, and all the variables that are negated are true. That means, none of the athletes in G_j are selected and all in B_j are selected, which means the team was weak for a skill s_j , a contradiction (since we had a valid team). This means the assignment described above is a satisfying assignment, thus, the SAT instance is satisfiable.

4) 18 pts

Suppose you are visiting a new city and planning to book hotels to stay at for D days. There are n hotels on a street (lined up next to each other as hotel 1, 2, ..., n) where the city government has regulated very low rates to encourage tourism. To prevent visitors from exploiting these rates for long stays, there is a mandated rule that you can **not** book a hotel for two days in a row. So you have to go stay in a different hotel each day. To reduce moving hassles, you want to move only to an adjacent hotel each day (e.g. you would not go directly to hotel 1 from hotel 3 etc.) The rates of the three hotels on day d are given by $R_1(d)$, $R_2(d)$, ..., $R_n(d)$ respectively ($d = 1, 2, \dots, D$). Finally, the availability of rooms is given by a boolean matrix A , where $A(i,d) = 1$ if Hotel i has a room available for booking on day d , and if not, $A(i,d) = 0$. Design an Integer Linear Program (ILP) to find your minimum hotel expenses for D days, given the constraints above.

a) Define the variables of your ILP. (2 pt)

$X_{i,d} = 1$ or 0 if you stay at hotel i on day d .

b) What is the objective function in your ILP? (4 pts)

Minimize $\sum_{i,d} X_{i,d} * R_i(d)$

c) What are the constraints in your ILP? (12 pts)

$\sum_i X_{i,d} = 1$ for all days d (exactly one hotel each day)

$X_{i,d} + X_{j,d+1} \leq 1$ for all days $d < D$, for all i, j s.t. $|i-j| \neq 1$
(consecutive stays only in adjacent hotels, not same or non-adjacent)

$X_{i,d} \leq A(i,d)$ (cannot book if room not available)

$X_{i,d} \in \{0,1\}$

5) 16 pts

A tech company produces two types of portable devices: **Tablets** and **Laptops**. Each Tablet brings a profit of \$120, and each Laptop brings a profit of \$200. The production of these devices requires resources from three departments: **Assembly**, **Testing**, and **Packaging**.

Producing one Tablet requires:

3 hours in Assembly

4 hours in Testing

1 hour in Packaging

Producing one Laptop requires:

4 hours in Assembly

3 hours in Testing

2 hours in Packaging

Each week, the company has:

240 hours of Assembly time

260 hours of Testing time

100 hours of Packaging time

Additionally, due to a contract, the company must produce **at least 10 Tablets** per week.

Complete the following steps describing your LP solution to this problem:

- a) Define decision variables. (4 points)
- b) Formulate the **objective function**. (2 points)
- c) Formulate the **constraints**. (10 points)

Answer Key:

1. Decision Variables:

Let x = number of Tablets to produce (2 points)

Let y = number of Laptops to produce (2 points)

2. Objective Function:

Maximize profit:

maximize $Z = 120x + 200y$ (2 points)

3. Constraints:

Assembly time:

$3x + 4y \leq 240$ (2 points)

Testing time:

$4x + 3y \leq 260$ (2 points)

Packaging time:

$x + 2y \leq 100$ (2 points)

Marketing requirement:

$x \geq 10$ (2 points)

Non-negativity:

$x \geq 0, y \geq 0$ (2 points)

6) 18 pts (3 pts each)

For each of the following 6 parts, either

- a. provide an example of a problem X that fits the description provided, or
- b. argue that such a problem does not exist (with a brief justification), or
- c. argue that it is unknown that such a problem exists (with a brief justification)

A) X is in NP and X cannot be solved in polynomial time

c. Unknown. X can be any NP-complete problem but only if $P \neq NP$

B) X is in NP-Hard, X has efficient certification, X is solvable

a. Any NP-complete problem (e.g. SAT, VC, IndSet)

C) X is not solvable, X is a decision problem

a. Halting problem

D) X is in NP, X is not solvable

b. all problems in NP are solvable

E) X is in NP, X is not in P, and X is not in NP-complete

c. Unknown. X can be any NP-intermediate problem only if $P \neq NP$

F) X has a polynomial time solution, X is an optimization problem

a. Any of the following: Shortest path, Max flow, etc.

Student ID:

Additional Space

Student ID:

Additional Space