Algorithm Design - Exam
27-01-2026, *Time: 120 minutes*
Prof. Aris Anagnostopoulos, Prof. Stefano Leonardi, Dr. Simone Fioravanti

Instructions:

- You are given the question sheet and 4 sheets for your answers. The problems are 4.
- Before you start, write on the top of each sheet (1) your full name, (2) your student ID (*matricola*) and (3) the sheet number (1, 2, 3, and 4).
- If you use additional sheets as rough draft (*brutta*) for calculations, etc., you must hand it, along with the rest of the exam. You can keep the question sheet.
- Hand in the sheets in order **without attaching the sheets with each other.**
- When your pages are full, you can ask for extra pages; make sure that you also write there your name, student id, and the sheet number (5, 6, etc.). You must hand all the sheets that you receive.
- The rooms are small so it may be tempting to copy. Note that if we catch you copying in any way (copying from another student, using your phone, etc.) you will be automatically disqualified.

Problem 1

Consider the following scenario. Due to large-scale flooding in a region, paramedics have identified a set of n injured people distributed across the region who need to be rushed to hospitals. There are k hospitals in the region, and each of the n people needs to be brought to a hospital that is within a half-hour's driving time of their current location (so different people will have different options for hospitals, depending on where they are right now). At the same time, one doesn't want to overload any one of the hospitals by sending too many patients its way. The paramedics are in touch by cell phone, and they want to collectively work out whether they can choose a hospital for each of the injured people in such a way that the load on the hospitals is balanced: Each hospital receives at most $\lceil n/k \rceil$ people.

(a) Give a polynomial-time algorithm that takes the given information about the people's locations and determines **whether** this is possible.

(b) Discuss its time complexity.

Problem 2

We are given a rod of integer length L. We assume that a rod of integer length $0 \le l \le L$ has a known value $v(l)$. The problem is to find the best way of cutting the rod, such that the sum of the values of the resulting rods is maximised.

(a) Show a dynamic programming algorithm that solves the problem.

(b) Provide the pseudo-code of an efficient implementation of the algorithm and discuss its time complexity.

Problem 3

Suppose you're helping to organize a summer sports camp, and the following problem comes up. The camp is supposed to have at least one counselor who's skilled at each of the n sports covered by the camp (baseball, volleyball, and so on). They have received job applications from c potential counselors. For each of the n sports, there is some subset of the c applicants qualified in that sport. The question is: For a given number $k < c$, is it possible to hire at most k of the counselors and have at least one counselor qualified in each of the n sports? We'll call this the EfficientRecruiting Problem. Show that EfficientRecruiting is NP-complete.

Problem 4

Consider the directed Max-Cut problem: given a directed graph $G = (V, A)$ with non-negative weights w_{ij} on the arcs, the problem is to find a cut $(S, V \setminus S)$ of the vertices, such that $\sum_{\substack{(i,j)\in A \\ i\in S, j\in T}} w_{ij}$ is maximized.

Show a randomized algorithm that achieves an expected approximation ratio of $\frac{1}{4}$.

Problem 1

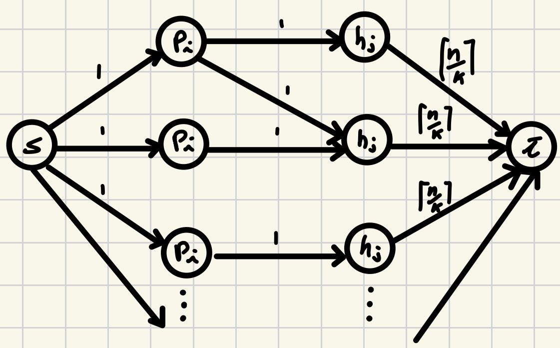
Consider the following scenario. Due to large-scale flooding in a region, paramedics have identified a set of n injured people distributed across the region who need to be rushed to hospitals. There are k hospitals in the region, and each of the n people needs to be brought to a hospital that is within a half-hour's driving time of their current location (so different people will have different options for hospitals, depending on where they are right now). At the same time, one doesn't want to overload any one of the hospitals by sending too many patients its way. The paramedics are in touch by cell phone, and they want to collectively work out whether they can choose a hospital for each of the injured people in such a way that the load on the hospitals is balanced: Each hospital receives at most $\lceil n/k \rceil$ people.

(a) Give a polynomial-time algorithm that takes the given information about the people's locations and determines **whether** this is possible.

(b) Discuss its time complexity.

K HOSPITAL n PEOPLE WITHIN 30 min

a) WE CONSTRUCT A FLOW NETWORK.
WE HAVE A SUPERSOURCE s WITH AN EDGE TOWARD EACH p WITH CAPACITY 1.
THEN EACH p_i IS LINKED WITH HOSPITALS WITHIN 30 min.
AND EACH h_j HAS CAPACITY $\lceil n/k \rceil$



CALCULATE THE MAX FLOW FROM s TO t:

- IF $v(f) = n$, EACH p_i IS IN A HOSPITAL
- IF $v(f) < n$, IT'S IMPOSSIBLE TO ASSIGN EACH p_i IN h_j SATISFYING THE CONSTRAINT $\lceil n/k \rceil$

b) $V = n + k + 2$
$E = n + m + k$ $\qquad O(n|E|)$

Problem 2

We are given a rod of integer length L. We assume that a rod of integer length $0 \leq l \leq L$ has a known value $v(l)$. The problem is to find the best way of cutting the rod, such that the sum of the values of the resulting rods is maximised.

(a) Show a dynamic programming algorithm that solves the problem.

(b) Provide the pseudo-code of an efficient implementation of the algorithm and discuss its time complexity.

a) $OPT(L) = \max_{1 \leq i \leq L} (v(i) + OPT(L-i))$ $\qquad OPT(0) = 0$

IF WE CUT IN POSITION i THE RESULT IS A CUT OF LENGTH i AND A CUT OF LENGTH $L-i$.

b) WE USE AN ARRAY $M[0..L]$ WHERE $M[j] = OPT(j)$

```
ROD-CUT

LET M[0..L]
M[0]=0
FOR j = 1 TO L:
    BEST = -∞
    FOR i = 1 TO L:
        BEST = max (BEST, v(i) + M[j-i])
    M[j] = BEST
RETURN M[L]
```

$O(L^2)$

Problem 3

Suppose you're helping to organize a summer sports camp, and the following problem comes up. The camp is supposed to have at least one counselor who's skilled at each of the n sports covered by the camp (baseball, volleyball, and so on). They have received job applications from c potential counselors. For each of the n sports, there is some subset of the c applicants qualified in that sport. The question is: For a given number $k < c$, is it possible to hire at most k of the counselors and have at least one counselor qualified in each of the n sports? We'll call this the EfficientRecruiting Problem. Show that EfficientRecruiting is NP-complete.

IT'S IN NP SINCE, GIVEN A SET OF K COUNSELORS, WE CAN CHECK IN POLYNOMIAL TIME THAT THEY COVER ALL THE SPORTS.

WE CAN PROVE THAT VERTEX COVER $\leq_P$ EFFREC

GIVEN A GRAPH G AND A NUMBER K, WE DEFINE A SPORT S_e FOR EACH EDGE AND A COUNSELOR C_u FOR EACH NODE u. C_u IS QUALIFIED IN S_e IFF S_e HAS AN END IN C_u.
THUS, G HAS A VERTEX COVER OF SIZE AT MOST K IFF THE INSTANCE OF EFFREC CAN BE SOLVED WITH AT MOST K COUNSELORS.

Problem 4

Consider the directed Max-Cut problem: given a directed graph $G = (V, A)$ with non-negative weights w_{ij} on the arcs, the problem is to find a cut $(S, V \setminus S)$ of the vertices, such that $\sum_{\substack{(i,j)\in A \\ i\in S, j\in T}} w_{ij}$ is maximized.

Show a randomized algorithm that achieves an expected approximation ratio of $\frac{1}{4}$.

INITIALIZE $S = T = \phi$
FOR EACH $v \in V$:
  PUT v RANDOMLY IN S OR IN T WITH PROB $\frac{1}{2}$
RETURN CUT (S, T)

E[CONTRIBUTION $e = (i, j)$] $= Pr[i \in S] \cdot Pr[j \in T] = \frac{1}{2} \frac{1}{2} = \frac{1}{4} w_{ij}$

LET OPT BE THE VALUE OF THE BEST CUT, AND SINCE ANY CUT CAN INCLUDE AT MOST ALL EDGES: OPT $\leq \sum_{(i,j)\in A} w_{ij}$

E[CUT'S VALUE] $= \frac{1}{4} \sum_{(i,j)\in A} w_{ij} \geq \frac{1}{4}$ OPT