Second Round Theoretical

Tasks

Swiss Olympiad in Informatics

March 7, 2020

Instructions

- Open your exam only after you have been told so. The examination starts for everyone at the same time and lasts 5 hours.
- With the exception of watches (no smart watches), no electronical devices are allowed on your table. Switch off your mobile phone.
- Start each task on a a separate sheet of paper and write your name on every sheet. Number your sheets and sort them before handing them in.
- Do not use pencil and do not write with red.
- Write legible.

Grading

The solutions will be graded according to similar criteria as the first theoretical round. The most important criteria are correctness and asymptotic running time. The quality of the description and the arguments asserting the correctness will also be taken into account.

You can always refer to some content of the SOI Wiki or 2H. You don't need to explain why e.g. Dijkstra works, but you should argue why and how it can be applied. In case of Dijkstra you should clearly state on which graph you run it on, and note that the edge weights are not negative.

The algorithm should be described in enough detail that it is easy to convert the description into a program. Also write down which data structures you would choose. Usually it is best to just write a short pseudocode.

To describe an algorithm, you should structure your solution as according to the following guideline:

1. Describe the idea for an algorithm that solves the problem.
2. Give pseudocode or explain how one would implement the algorithm.
3. Argue about the correctness of the approach.
4. Indicate asymptotic running time and memory usage.

If some part of your solution can be used for multiple subtasks, it suffices to write it down only once and refer to it from other parts. Note, however, that a general algorithm might be able to solve the previous subtasks, but not necessarily optimally.

Evaporation

Mouse Binna is super happy to be able to compete in the IOI 2020 in Singapore. However, once arrived, she soon realised Singapore is way too hot for a little mouse like her. In order not to die of dehydration, Mouse Binna decided to buy a lot of glasses of water, namely n. The i-th glass contains a_i milliliters of water. Unfortunately, because of the hot weather, every minute one milliliter of water evaporates from every glass. As Mouse Binna does not want to spend all her time going to the supermarket to buy water, at the start of every minute Mouse Binna decided she may pour all the water from one glass into another glass. Help Mouse Binna maximize the amount of water after n minutes.

Formal Description Given is a list of n non-negative integers $a_0, \ldots, a_{n-1}$. Every minute, the following happens: you may set a_i to $a_i + a_j$ and a_j to 0 (for i and j of your choice), and then all integers $a_i > 0$ become $a_i - 1$. Find the maximal sum of all integers after n minutes.

Subtask 1: Solving an example (10 points)

Given is the list: $\{1, 2, 3, 4\}$. Which is the maximum amount of water which is still present after $n = 4$ minutes? List the operations and the time you do those actions in order to get this value at the end. You do not need to justify your answer.

Subtask 2: All the same (20 points)

We now assume all glasses contain in the beginning exactly k milliliter of water ($a_i = k$ for all $0 \leq i < n$). Construct an algorithm that computes the maximal value in this special case (and prove its correctness).

Subtask 3: Permutations (30 points)

This time, all integers a_i form a permutation of the numbers $\{1, 2, \ldots, n\}$. Construct an algorithm in this case (and prove its correctness).

Subtask 4: General case (40 points)

There is no more restrictions on the values of the a_i's. Construct an algorithm for this general case.

Number Hunt

Mouse Stofl is on a number hunt where the goal is to find a missing number. There is a secret table, unknown to Stofl, of $2^n - 1$ unique integers from 0 to $2^n - 1$. Hence exactly one number is missing. Help Mouse Stofl to figure out the missing number in order to win the number hunt.

Any number in the secret table can be written as a sequence of n bits. E.g. $6_{(10)} = 4 + 2 = 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 110_{(2)}$ or $13_{(10)} = 8 + 4 + 1 = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 1101_{(2)}$. The 0-th bit is the coefficient of 2^0 (i.e. 1 if the number is odd and 0 if it is even), the 1-th bit is the coefficient of 2^1, and so on.

There are $(2^n - 1) \cdot n$ possible quests available for Mouse Stofl. Solving the quest $q(i, j)$ reveals him the j-th bit of the i-th number. In the table below solving $q(5, 2)$ reveals 1 and solving $q(0, 2)$ reveals 0. Quests are solved one by one, so Stofl can decide which quest to do next based on the results of the previous quests.

Help Stofl to win the number hunt with the minimal number of quests.

Num	Bits		
$a_0 = 3$	0	1	1
$a_1 = 1$	0	0	1
$a_2 = 6$	1	1	0
$a_3 = 7$	1	1	1
$a_4 = 0$	0	0	0
$a_5 = 4$	1	0	0
$a_6 = 2$	0	1	0

Example of a secret table

Num	b_2	b_1	b_0
a_0	$q(0,2)$	$q(0,1)$	$q(0,0)$
a_1	$q(1,2)$	$q(1,1)$	$q(1,0)$
a_2	$q(2,2)$	$q(2,1)$	$q(2,0)$
a_3	$q(3,2)$	$q(3,1)$	$q(3,0)$
a_4	$q(4,2)$	$q(4,1)$	$q(4,0)$
a_5	$q(5,2)$	$q(5,1)$	$q(5,0)$
a_6	$q(6,2)$	$q(6,1)$	$q(6,0)$

Table of quests Stofl can solve

Formal description Let $a_0, a_1, \ldots, a_{2^n-1}$ be a permutation of the numbers $\{0, 1, \ldots, 2^n - 1\}$. You can make queries $q(i, j)$ for $0 \le i \le 2^n - 2$ and $0 \le j < n$ to get access to the j-th bit of the i-th number (note that you cannot query the $(2^n - 1)$-th number). Queries are made one by one, so you can decide which query to make next based on the results of the previous queries. Find out the value of a_{2^n-1} with the minimal number of queries.

Subtask 1: Analyze Stofls Table (10 points)

Stofl has already solved some of the quests and has obtained the table below:

Num	b_2	b_1	b_0
a_0		1	0
a_1	1		
a_2	1		1
a_3		1	0
a_4			0
a_5	1		1
a_6			

Solve the following tasks (no justification required):

- For each element of the list a_i, $0 \le i \le 6$, write down the set of all possible values.
- The missing number can be determined uniquely by a single quest. Which quest and how do you find the number afterwards?

Subtask 2: Optimal Algorithm (90 points)

Design an algorithm that solves the problem optimally. You are given n and should solve some quests in order to find out the missing number. Note that you can adapt your strategy based on the results of the previous quests.

Optimize first for the number of quests. Give the exact formula for the number of quests you need in the worst case (e.g. $2^n + n + 13$ or $\frac{1}{2}n \cdot (n-1)$). Then optimize for (asymptotic) running time, and then for (asymptotic) space usage.

Binnabike

Mouse Binna just arrived at Singapore for the IOI 2020. She is currently at her hotel and would like to go to the opening ceremony by bike, as it is quite far away. Mouse Binna rents a Binnabike to travel through the city. It functions as follows:

- As Singapore wants more people to use eco-friendly transport, the government decided that the first d minutes riding a bike from any station will always be free – this can be done multiple times per day!
- After the first d minutes, one dollar has to be paid per minute.
- The bike always has to be brought back to an official station

In Singapore there are n crossings connected by m streets. For each street you know how many minutes Binna needs to travel along it; this time is given as an integer > 0. She also knows that k of those crossing are also Binnabike stations where bikes can be picked up and brought back.

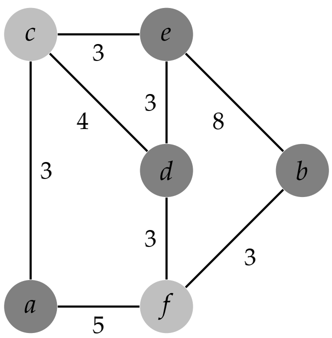
Binna finds out that station a is the nearest from her hotel and station b is the nearest to the opening ceremony. As the latter starts at time t, Binna naturally wants to arrive at station b before t. Additionally, she would like to spend as little money as possible along the way. Help her!

Formal Description Given is a graph with n vertices and m edges. The edges are weighted by an integer > 0. k of the vertices are called stations. There is a discount parameter d. If the time to travel from station u to station v is w, the cost equals $\max(w - d, 0)$, and this discount can be used multiple times. Find the minimal cost to travel from station a to b such that the time required is at most t.

It is guaranteed that there is at least one path to travel from a to b in time t.

Limits For optimizing your algorithm, you can assume that n and t are very large and d and k are fairly small. In particular, you can assume $d \cdot k$ (and thus d and k) is smaller than t and that k^3 is smaller than n. The edge weights are integers > 0 that can be arbitrarily large.

Subtask 1: Solving an example (5 points)



Give the path Binna should take in order to minimize the cost for the graph above, with $d = 3$ and $t = 14$. The $k = 4$ stations are a, b, d, e. Give the amount of dollars she has to spend. No justification is needed.

Subtask 2: Unlimited time (35 points)

Because Binna thinks the opening ceremony is pretty much the same every year, she finds it acceptable to be late. Construct an algorithm which calculates the minimum cost given that Binna has unlimited time.

Subtask 3: General case (60 points)

Unfortunately, the Swiss leaders ordered Binna to meet them at the opening ceremony before it starts, so Binna will have to arrive sooner than expected. There are no restrictions anymore. Describe your algorithm.

Venice

Unfortunately, Venice is flooded due to sea level rise and closed for visitors. Because Mouse Stofl always wanted to visit Venice, but has never had the opportunity to do so, he is very sad. So Mouse Binna started to reconstruct the city. She has already rebuilt n sights and dug channels between them. As it is expensive to dig channels, she has only built the channels needed so that all sights are connected.

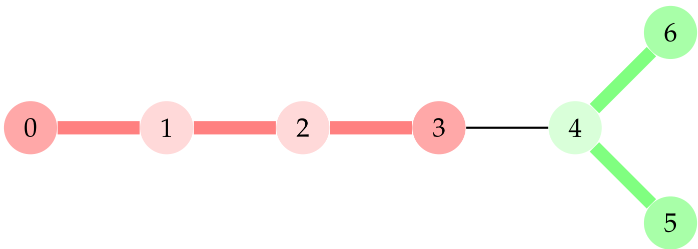
The most important thing in Venice are the gondolas: each gondola operates on a path between a pair of sights and has a stop at each sight it passes by.

Building gondolas is not cheap either, so Mouse Binna wants to design the gondola routes so that the least number of gondolas is required while still having a stop of some route at each sight. Help her to determine how many routes are needed and how they should be chosen.

Formal description Given is a graph consisting of the sights and the channels connecting them. It is guaranteed that the graph is a tree. In particular, this means that any two sights a and b are connected by a unique path.

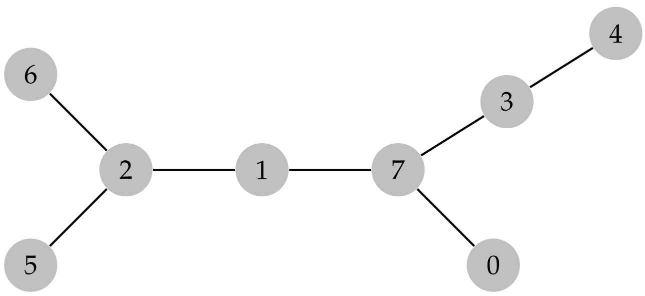
A gondola schedule is a list of pairs (a_i, b_i), where the i-th gondola operates between the sights a_i and b_i. A gondola schedule is valid if for every sight there exists some gondola route that has a stop at it. A gondola schedule is optimal if it is valid and there is no valid gondola schedule that uses less gondolas.

Example We have $n = 7$ and the list of channels is $\{(0, 1), (1, 2), (2, 3), (3, 4), (4, 5), (4, 6)\}$. Then a valid (and optimal) gondola schedule is to operate a gondola between 0 and 3 and another one between 5 and 6. This schedule is depicted below.



Subtask 1: Solve an example (10 points)

There are 8 sights in Venice that are connected by channels as depicted in the figure below. Find an optimal gondola schedule and justify its optimality.



Subtask 2: More gondolas (25 points)

Mouse Binna is happy if you find a gondola schedule that is almost optimal. More precisely, if an optimal gondola schedule requires k gondolas, then Mouse Binna wants you to find a valid gondola schedule that requires at most $2k$ gondolas (we call such a gondola schedule almost optimal). Design an algorithm that computes such an almost optimal gondola schedule and justify its correctness.

Subtask 3: Optimal schedule (65 points)

Building gondolas is far more expensive than Mouse Binna hoped. Hence, she is now only happy if you find an optimal gondola schedule. Design an algorithm that computes an optimal gondola schedule and justify its correctness.