The Coding DEI

Programming Contest Training

What is a programming contest?

Challenge:

- Solve well-defined problems by writing computer programs under specified limits

Solve well-defined problem…

Input

The input consists of:

- One line with the integer n ($1 \leq n \leq 2 \cdot 10^5$), the number of balloons and gas canisters.
- One line with n integers $c_1, \ldots, c_n$ ($0 \leq c_i \leq n$ for each i), the amounts of helium in the gas canisters, in decilitres.

Output

If it is possible to fill all the balloons without any exploding, output the maximum fraction f such that every balloon can be filled to at least f of its capacity. Otherwise, output "`impossible`".

Your answer should have an absolute or relative error of at most 10^{-6}.

Sample Input 1

```
6
6 1 3 2 2 3
```

Sample Output 1

```
0.6
```

Sample Input 2

```
2
2 2
```

Sample Output 2

```
impossible
```

For NWERC 2018, the organis
balloons. Instead of buying ba
every integer size from 1 up
decilitres.

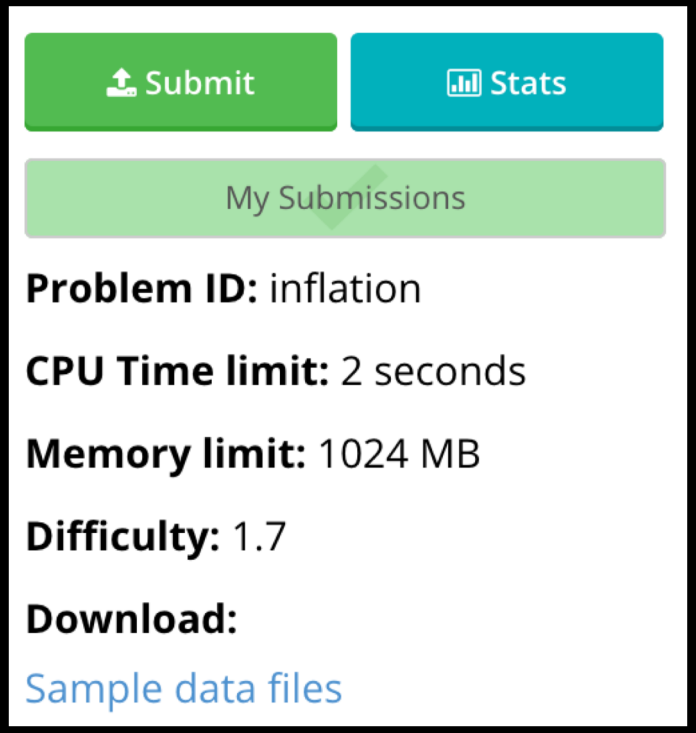
To avoid inflating the balloon
gas canisters. Each canister c
balloon (it is not possible to di

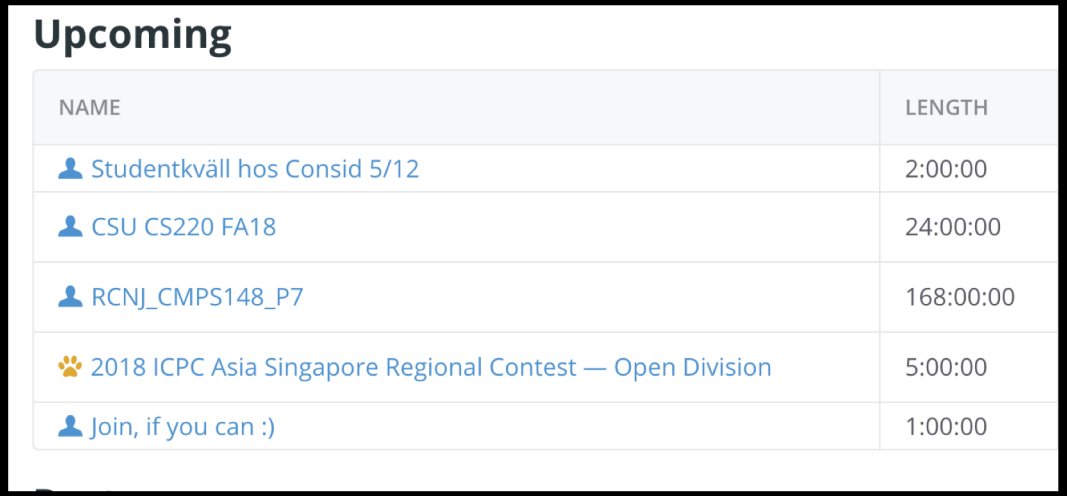
Unfortunately the gas caniste
may even be empty! To make
balloons smartly.

The organisers want to assign
least (relative to its capacity)
such (minimum) fraction that

Balloons filled beyond their capacity will explode. Explosions are upsetting and must be avoided.

… under specific limits…





..by writing computer programs...

Submission contains 1 file: download zip archive

FILENAME	FILESIZE	SHA-1 SUM
l.py	384 bytes	8b08bb9ca602ecd04255c47e0eb83da1998e5835

Edit and resubmit this submission.

l.py

```
1  import sys
2
3  n = int(sys.stdin.readline());
4
5  list = [];
6
7  for i in sys.stdin.readline().split():
8      list.append(int(i));
9
10 list.sort();
11
12 min = 2.0;
13 ratio = 0.0;
14 num = 1.0;
15
```

MANY LANGUAGES:
C, C++, Java, Python,..

Many of them!

Problems from Northwestern Europe Regional Contest (NWERC) 2018

SOLVED TRIED UNTRIED

	SUBMISSIONS				USERS					
NAME	TOTAL	ACC.	RATIO	FASTEST	TOTAL	ACC.	RATIO	DIFFICULTY		
Access Points	69	24	35%	0.02	32	22	69%	5.9		
Brexit Negotiations	168	71	42%	0.16	82	63	77%	4.8		
Circuit Board Design	146	47	32%	0.00	48	41	85%	5.3		
Date Pickup	21	7	33%	0.62	7	5	71%	6.9		
Equality Control	212	42	20%	0.03	51	32	63%	7.3		
Fastest Speedrun	57	16	28%	0.66	15	10	67%	8.3		
Game Design	163	53	33%	0.00	51	44	86%	5.3		
Hard Drive	297	114	38%	0.01	116	104	90%	3.3		
Jinxed Betting	154	34	22%	0.02	48	32	67%	6.0		
Kleptography	129	114	88%	0.00	110	109	99%	1.6		

Who?

Target:

- People with interest in algorithms and coding
- Some challenges are only for students (e.g., ICPC)

- Single user
- Team work

Where?

- Many are online

Ongoing

NAME	REMAINING	LENGTH	START TIME	
AvaSE-NOV-2018	2:09:14	336:00:00	2018-11-16 23:59:00 CET	
hello	2:10:14	5:00:00	19:00:00 CET	
2015 ACM ICPC Singapore Regional	22:10:14	168:00:00	2018-11-24 20:00:00 CET	Join the contest
PSU FA18 12	98:10:14	168:00:00	2018-11-28 00:00:00 CET	Join the contest
CSCI 371 Post Season Practice 3	112:25:14	160:00:00	2018-11-28 22:15:00 CET	
CSCI 499 Final Set	120:10:14	168:00:00	2018-11-28 22:00:00 CET	Join the contest
FRA UAS Exercises Week7	132:10:14	160:00:00	2018-11-29 18:00:00 CET	Join the contest
JMU F18 Week 13	144:10:14	145:30:00	20:30:00 CET	Join the contest
Frankfurt UAS Exercises Week 7	156:10:14	168:00:00	10:00:00 CET	Join the contest

- The most important are offline

Some challenges:

International Collegiate Programming Contest (ICPC)

Southwestern Europe Regional Contest (SWERC)

Olympiad in Informatics (only high school)

Google Code Jam

Facebook Hacker Cup

ICFP Programming Contest

Many websites: Codeforces, Kattis, Project Euler, UVa Online Judge, …

Skills

Algorithmic problem solving

- Understand the problem
- Solve the problem
- Design an algorithm

Practical coding

- Implementation
- Debug

Team work

- Strategy
- Collaboration

Training

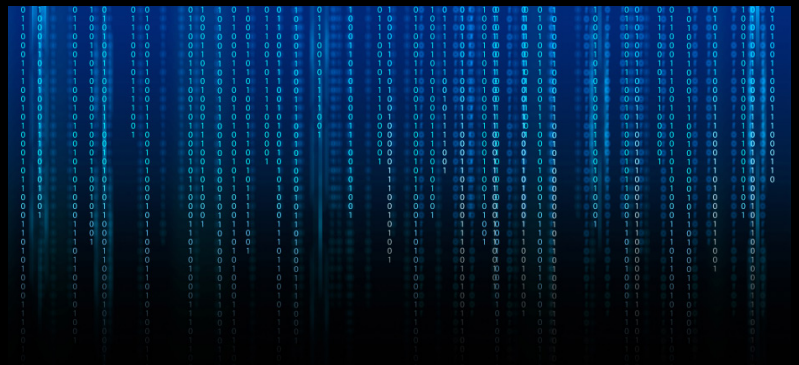
- Programming challenges are not easy!
- You need to train, exercise, study,… like all sports

- Some skills:
  - Algorithmic problem solving: data structures, dynamic programming, recursion, graph problems,…
  - Practical coding: programming, debug, standard libraries,…
  - Team work: specialization, sharing one keyboard,…

The Coding DEI

The Coding DEI is a lab for training your skills in programming contests

- Play with programming challenges
- Solve problems
- Discuss solutions and implementations



The Coding DEI (2)

The Coding DEI IS NOT a course

The Coding DEI IS NOT mandatory

The Coding DEI DOES NOT give credits

The Coding DEI IS a voluntary activity

The Coding DEI IS an informal activity

The Coding DEI (3)

- Everybody is welcome!
- If you like programming challenges but without expertise, you are welcome.
  - You will learn
- If you like programming challenges and you have a lot of experience, you are welcome.
  - You will still learn
  - You will help your colleagues

Why to attend? (1)

Because you like

- Solving problems
- Coding
- Algorithms and data structures
- Challenges
- Team work

Why to attend? (2)

Because it is fun to

- Compete with your friends
- Create a team
- Participate in real programming contests
- Solve a problem faster than a professor

Why to attend? (3)

Because you improve your curriculum

- New coding/algorithmic skills
- That can help in preparing exams…
- … and in solving hiring tests.

- Many companies (Google, Facebook,…) ask to solve computational problems in the hiring interviews

Next meetings

- Wednesday April 17, 16.30 @ TE
- Wednesday May 15, 16.30 @ TE
- Wednesday June 5, 16.30 @ TE

- About 120 minutes per meeting

www.dei.unipd.it/thecodingdei

- Communication via Slack https://thecodingdei.slack.com/
- Create an account in open.kattis.com (and start playing)
- You can use your laptop