

Assignment 1 - Trik, Guess the Data Structure, Flexible Spaces

COMP 321 - Programming Challenges

LE, Nhat Hung

McGill ID: 260793376

Date: January 22, 2020

Due date: January 29, 2020

Prof. David Meger

Winter 2020

Note:

McGill got my name wrong. On MyCourses, it appears as Nhat Le, instead of Hung Le. Enrollment services will fix it soon.

Question 1: Trik

Kattis PROBLEMS CONTESTS RANKLISTS JOBS HELP Search Kattis Submit Hung Le Score: 8.2, Rank: 33582

Submission

ID	DATE	PROBLEM	STATUS	CPU	LANG
TEST CASES					
5218234	13:39:45	Trik	✓ Accepted	0.05 s	Python 3
✓✓✓✓✓					

Submission contains 1 file: [download zip archive](#)


FILENAME	FILESIZE	SHA-1 SUM	
trik.py	627 bytes	77d751b7cf6e93953f5966b3a5eabb2729d71652	download

[Edit and resubmit this submission.](#)

trik.py

```
1 #!/usr/bin/python3
2
3 import sys
4
5 def trik(moves):
6     # 3 cups, 1 means has ball, 0 means no ball
7     cups = [1,0,0]
8
9     # Execute each move by swapping cups
10    for move in moves:
11        if move == 'A':
12            cups[0], cups[1] = cups[1], cups[0]
13
14        elif move == 'B':
15            cups[1], cups[2] = cups[2], cups[1]
16
17        else: # move == 'C':
18            cups[0], cups[2] = cups[2], cups[0]
19
20    # Find index of cup with ball, return that index + 1
```


Question 2: Guess the Data Structure

 **Kattis**
PROBLEMS CONTESTS RANKLISTS JOBS HELP


Search Kattis

Submit

Score: 8.2, Rank: 33582



Submission

ID	DATE	PROBLEM	STATUS	CPU	LANG
TEST CASES					
5218266	13:51:31	I Can Guess the Data Structure!	Accepted	0.16 s	Python 3
					

Submission contains 1 file: [download zip archive](#)

FILENAME	FILESIZE	SHA-1 SUM	
guessthedatastructure.py	2197 bytes	0de7e443e409b143c5b09045161810fa74df56bb	download


[Edit and resubmit](#) this submission.

guessthedatastructure.py

```
1 #!/usr/bin/python3
2
3 import sys
4
5 def main():
6     # Convert stdin into list
7     inp = [*sys.stdin]
8
9     # For each line of stdin
10    for i, line in enumerate(inp):
11        # Sanitize line
12        line = line.strip().split()
13
14        # If line is one integer, then it's the start of a series of operations
15        if len(line) == 1:
16            # Extract operations
17            num_ops = int(line[0])
18            ops = inp[i + 1 : i + 1 + num_ops]
19
20            # Print data structure
```

Support


Question 3: Flexible Spaces

 **Kattis**
PROBLEMS CONTESTS RANKLISTS JOBS HELP










Search Kattis

Submit

Score: 8.2, Rank: 33582



Submission

ID	DATE	PROBLEM	STATUS	CPU	LANG
TEST CASES					
5218302	14:02:59	Flexible Spaces	Accepted	0.05 s	Python 3
        					

Submission contains 1 file: [download zip archive](#)

FILENAME	FILESIZE	SHA-1 SUM	
flexible.py	842 bytes	90730ea46a43952cb39d82bcbbedbcee4d9df31a	download

[Edit and resubmit](#) this submission.

flexible.py

```
1 #!/usr/bin/python3
2
3 import sys
4
5 # Convert stdin to list, sanitize each element
6 inp = [line.strip().split() for line in sys.stdin]
7
8 # Extract variables
9 w = int(inp[0][0])
10 p = int(inp[0][1])
11
12 # Add w to 1 for efficiency
13 l = [int(loc) for loc in inp[1]] + [w]
14
15 # Initialize output widths as all values in l
16 widths = [1]
17
18 # Add to output widths all possible widths that are not yet in the output widths list
19 for i in range(len(l)):
20     for j in range(1 + 1 len(l)):
```

Support