A. Requirements

Code (90%)

You can write your code in Java, Python, C, or C++. The *time limit* may vary among different languages, depending on the performance of the language. Your code must be a complete excutable program instead of only a function. We guarantee test data strictly compliance with the requirements in the description, and you do not need to deal with cases where the input data is invalid.

Libraries in this assignment:

- For C/C++, you can only include standard library.

- For Java, you can only `import java.util.*`

- For Python, you can only import standard library. In other words, you cannot import libraries such as `numpy`.

We provide an example problem to illustrate the information above better.

Report (10%)

You also need to write a report in `pdf` type to explain the following:

- What are the possible solutions for the problem?

- How do you solve this problem?

- Why is your solution better than others?

Please note that the **maximum** number of pages allowed for your report is **5 pages**.

Remember that the report is to illustrate your thinking process. Keep in mind that your report is supposed to show your ideas and thinking process. We expect clear and precise textual descriptions in your report, and we do not recommend that you over-format your report.

B. Example Problem: A + B Problem

Description

Given 2 integers A and B, compute and print A + B

Input

Two integers in one line: A, and B

Output

One integer: A + B

Sample Input 1

```
1 2
```

Sample Output 1

```
3
```

Problem Scale & Subtasks

For 100% of the test cases, $0 \le A, B \le 10^6$

Solutions

Java

```java
import java.util.*;

public class Example {
    public static void main(String[] args) {
        int a, b;
        Scanner scanner = new Scanner(System.in);
        a = scanner.nextInt();
        b = scanner.nextInt();
        scanner.close();
        System.out.println(a + b);
    }
}
```

Python

```python
AB = input().split()
A, B = int(AB[0]), int(AB[1])
print(A + B)
```

C

```c
#include <stdio.h>

int main(int argc, char *argv[])
{
  int A, B;
  scanf("%d%d", &A, &B);
  printf("%d\n", A + B);
  return 0;
}
```

C++

```cpp
#include <iostream>

int main(int argc, char *argv[])
{
  int A, B;
  std::cin>> A >> B;
  std::cout<< A + B << std::endl;
  return 0;
}
```

C. Submission

After finishing this assignment, you are required to submit your code to the Online Judge System (OJ), and upload your `.zip` package of your code files and report to BlackBoard.

C.1 Online Judge

Once you have completed one problem, you can submit your code on the page on the Online Judge platform (oj.cuhk.edu.cn, campus only) to gain marks for the code part. You can submit your solution of one problem for **no more than 80 times**.

After you have submitted your program, OJ will test your program on all test cases and give you a grade. The grade of your latest submission will be regarded as the final grade of the corresponding problem. Each problem is tested on multiple test cases of different difficulty. You will get a part of the score even if your algorithm is not the best.

Note: The program running time may vary on different machines. Please refer to the result of the online judge system. OJ will show the time and memory limits for different languages on the corresponding problem page.

If you have other questions about the online judge system, please refer to OJ wiki (campus network only). If this cannot help you, feel free to contact us.

C.2 BlackBoard

You are required to upload your **source codes and report** to the BlackBoard platform. You need to name your files according to the following rules and compress them into `A2_<Student ID>.zip` :

```
A2_<Student ID>.zip
|-- A2_P1_<Student ID>.java/py/c/cpp
|-- A2_P2_<Student ID>.java/py/c/cpp
|-- A2_Report_<Student ID>.pdf
```

For Java users, **you don't need to consider the consistency of class name and file name**.

For example, suppose your ID is 123456789, and your problem 1 is written in `Python`, problem 2 is written in `Java` then the following contents should be included in your submitted `A2_123456789.zip`:

```
A2_123456789.zip
|-- A2_P1_123456789.py
|-- A2_P2_123456789.java
|-- A2_Report_123456789.pdf
```

C.3 Late Submissions

Submissions after Nov 8 2023 23:59:00(UTC+8) would be considered as LATE.

The LATE submission page will open after deadline on OJ.

Submisson time = max{latest submisson time for every problem, BlackBoard submisson time}

There will be penalties for late submission:

- 0–24 hours after deadline: final score = your score$\times 0.8$
- 24–72 hours after deadline: final score = your score$\times 0.5$
- 72+ hours after deadline: final score = your score$\times 0$

FAQs

Q: I cannot access to Online Judge.

A: First, please ensure that you are using the campus network. If you are not on campus, please use the university VPN. Second, please delete cookies and refresh browser or use other browser. If you still cannot access to Online Judge, try to visit it via the IP address `10.26.200.13`.

Q: My program passes samples on my computer, but not get AC on OJ.

A: Refer to OJ Wiki Q&A

Authors

If you have questions for the problems below, please contact:

- Problem 1. Bingwei Zhang: bingweizhang@link.cuhk.edu.cn
- Problem 2. Yohandi: yohandi@link.cuhk.edu.cn

CSC3100 Data Structures Fall 2023

Programming Assignment 2

Due: Nov 8 2023 23:59:00

Assignment Link: http://oj.cuhk.edu.cn/contest/csc310023falla2

Access Code: 7d2x0Zs8

1 Battle Game (40% of this assignment)

1.1 Description

Imagine a group of n players that are located in different floors. The i-th player (for each i such that $1 \le i \le n$, where i is an integer) is characterized by the following:

- p_i, a unique starting floor level
- h_i, a starting HP (Health Points) value
- d_i, a direction of movement, either U indicating that the i-th player is going upward or D indicating that the i-th player is going downward

All players move simultaneously at the same speed in their respective directions. When two players meet, they engage in a battle. The player with the lower health is eliminated **instantaneously**, while the other continue to move in the same direction at the same speed but with a reduced HP by 1. In a case where two players have identical HP, they are both eliminated simultaneously after the battle.

Your task is to determine the final HP of the surviving players.

1.2 Input

The first line of input contains an integer n, representing the number of players.

The next n lines follow.

The i-th line contains three values: p_i, h_i, d_i - an integer representing the starting floor level for the i-th player, an integer representing the HP value for the i-th player, and a character representing the direction of movement for the i-th player.

1.3 Output

For each surviving player, output their remaining HP on a separate line, in the order of their input.

Sample Input 1

```
5
5 11 U
4 15 U
3 17 U
2 77 U
1 30 U
```

Sample Output 1

```
11
15
17
77
30
```
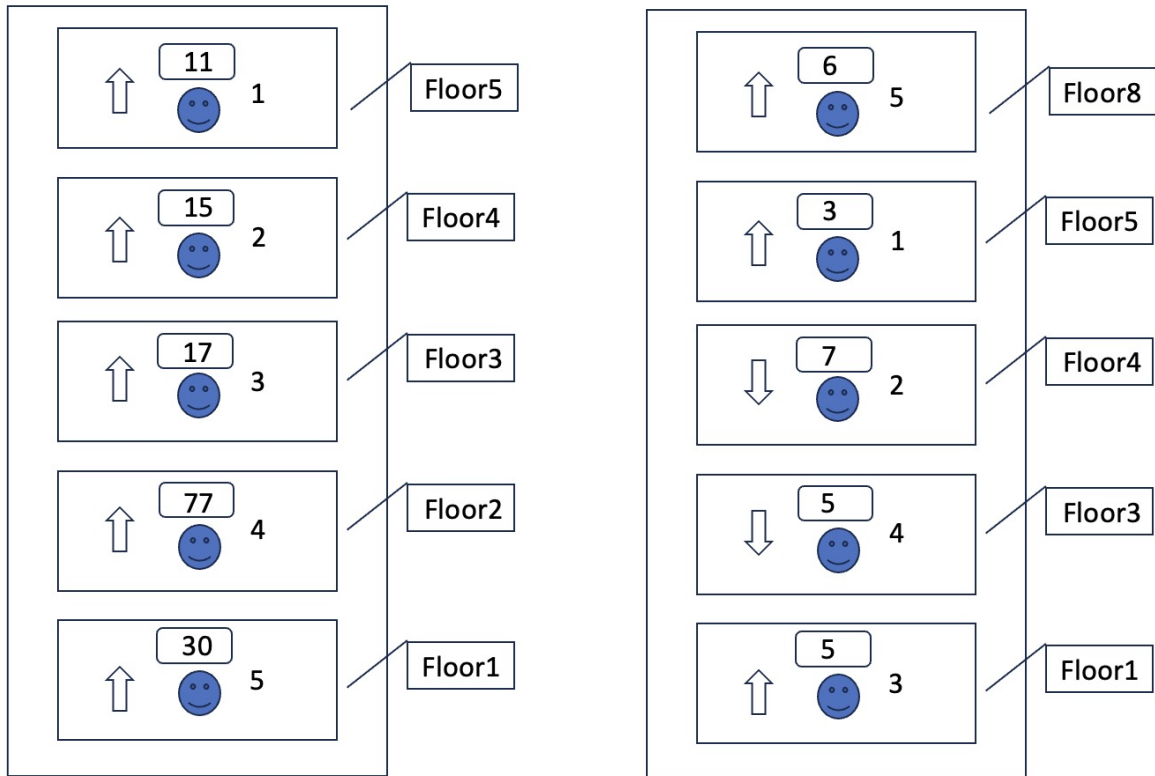
Sample Input 2

```
5
5  3  U
4  7  D
1  5  U
3  5  D
8  6  U
```

Sample Output 2

```
3
7
6
```

The following figures show the visual representations for both input samples in their initial positions, respectively.

Problem Scale & Subtasks

For 100% of the test cases:

- $1 \le n \le 10^6$

- $1 \le p_i \le 10^8$ (for each i such that $1 \le i \le n$)

- $1 \le h_i \le 100$ (for each i such that $1 \le i \le n$)

- $d_i \in \{\texttt{U}, \texttt{D}\}$ (for each i such that $1 \le i \le n$)

Test Case No.	Constraints
1-4	$n \le 1\,000$
5-8	$n \le 10\,000$
9-10	No additional constraints

5

2 Buried Treasure (50% of this assignment)

2.1 Description

Jack Sparrow, the legendary pirate of the Seven Seas, sets sail to an inhabited island in search of a buried treasure. Guided by his map, he believes the treasure lies within a weird-looking trench. The trench stretches across a width of n. For any given point from $i-1$ to i on the trench's x-coordinate, the depth is represented by d_i (for each i such that $1 \leq i \leq n$, where i is an integer).

Jack is wondering about the size of the largest treasure that could possibly fit inside the trench. Wanting to maximize his haul, he turns to you, a trusted member of his crew, to make the calculations. By largest, Jack means the maximum area – the product of width and height – of the **rectangular** treasure chest that can be buried within the trench's confines. For example, the following figure shows the largest possible treasure that can fit in a trench with $n = 8$ and $d = [6, 2, 5, 4, 5, 1, 4, 4]$.
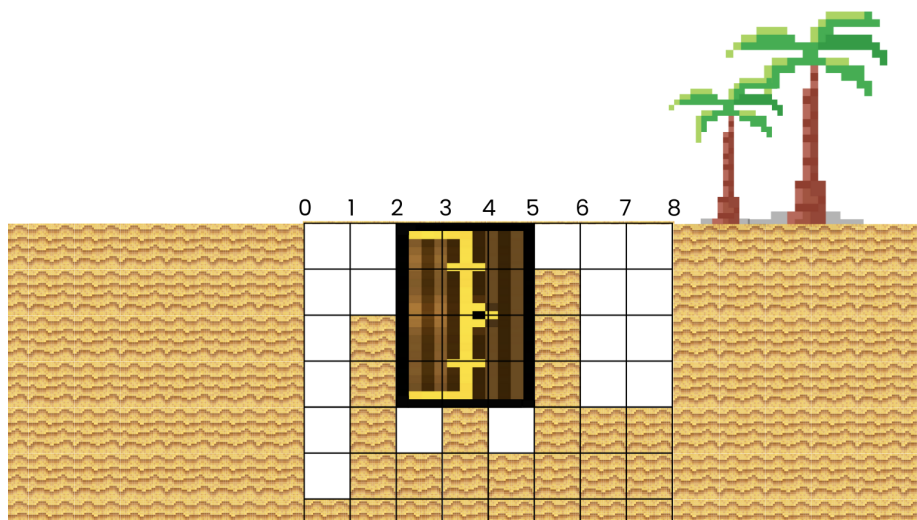
Could you give these calculations a look for our legendary pirate?

2.2 Input

The first line contains an integer T, representing the number of Jack Sparrow's queries.

The descriptions of T queries follow.

Each query is described in the following format:

- The first line contains an integer n representing the width of the trench.

- The second line contains n integers separated by a space representing the depths of the trench.

2.3 Output

Output T lines. The j-th line contains the answer to the j-th query.

Sample Input 1

```
2
8
6 2 5 4 5 1 4 4
3
2 7 4
```

Sample Output 1

```
12
8
```

Problem Scale & Subtasks

For 100% of the test cases:

- $1 \le T \le 10$

- $1 \le n \le 10^6$

- $1 \le d_i \le 10^6$ (for each i such that $1 \le i \le n$)

- The sum of n over all queries, denoted as N, does not exceed 10^6 in each test case.

Test Case No.	Constraints
1-2	$n \le 10$
3-4	$n \le 1\,000$
5-8	$N \le 10\,000$
9-10	No additional constraints

Hint

Hint 1: For C/C++ and Java users, an `int` type stores integers range from -2,147,483,648 to 2,147,483,647. It may be too small for this problem. You need other data types, such as `long long` for C/C++ and `long` for Java. They store integers ranging from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807. Use `scanf("%lld",&n)` for C, `cin>>n` for C++ and `n = scanner.nextLong()` for Java to output the answer query. And the other operations for `long` and `long long` are quite same as `int`.

Hint 2: **S**ailing through the vast expanse of the Seven Seas, Jack Sparrow found himself drawn to an intriguing map. **T**antalized by the prospect of hidden riches, he anchored near a mysterious island with an unusual trench. **A**s he examined the trench's varying depths, he pondered the size of the treasure that might be buried within. **C**arefully, he sought the counsel of a trusted crew member to calculate the chest's potential dimensions. **K**nowing the importance of this task, the crew member took to the challenge, hoping to maximize the treasure for their legendary captain.