STMC Coding Team Training

Lesson 9: HKOI question training

Tsai Yun Chen

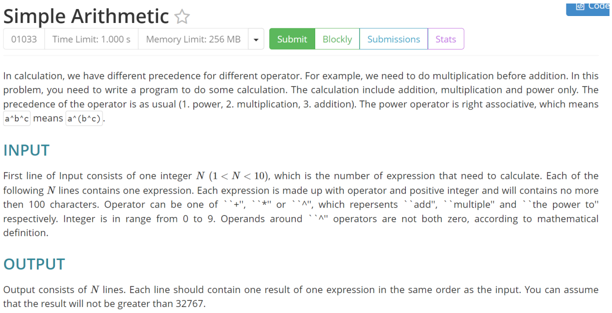
May 5, 2023

Goal today

Today we will practice several questions on the platform of HKOI.

- 01033 Simple Arithmetic
- 01015 Parenthesis Balance
- 01077 Course Registration
- 01017 Car Sorter
- Challenge: M1313 Bookstack

Simple Arithmetic


Code

Simple Arithmetic ☆

01033 | Time Limit: 1.000 s | Memory Limit: 256 MB | Submit | Blockly | Submissions | Stats

In calculation, we have different precedence for different operator. For example, we need to do multiplication before addition. In this problem, you need to write a program to do some calculation. The calculation include addition, multiplication and power only. The precedence of the operator is as usual (1. power, 2. multiplication, 3. addition). The power operator is right associative, which means `a^b^c` means `a^(b^c)`.

INPUT

First line of Input consists of one integer N $(1 < N < 10)$, which is the number of expression that need to calculate. Each of the following N lines contains one expression. Each expression is made up with operator and positive integer and will contains no more then 100 characters. Operator can be one of ``+", ``*" or ``^", which repersents ``add", ``multiple" and ``the power to" respectively. Integer is in range from 0 to 9. Operands around ``^" operators are not both zero, according to mathematical definition.

OUTPUT

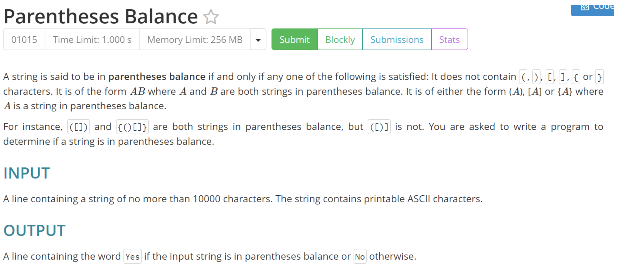
Output consists of N lines. Each line should contain one result of one expression in the same order as the input. You can assume that the result will not be greater than 32767.

Figure 1: Simple Arithmetic(Source)

Hint

- Recall the topic we convered last lecture
- slightly modify the code to obtain the solution

Parenthesis Balance


Parentheses Balance ☆

01015 | Time Limit: 1.000 s | Memory Limit: 256 MB

A string is said to be in **parentheses balance** if and only if any one of the following is satisfied: It does not contain `(`, `)`, `[`, `]`, `{` or `}` characters. It is of the form AB where A and B are both strings in parentheses balance. It is of either the form (A), $[A]$ or $\{A\}$ where A is a string in parentheses balance.

For instance, `([])` and `{()[]}` are both strings in parentheses balance, but `([)]` is not. You are asked to write a program to determine if a string is in parentheses balance.

INPUT

A line containing a string of no more than 10000 characters. The string contains printable ASCII characters.

OUTPUT

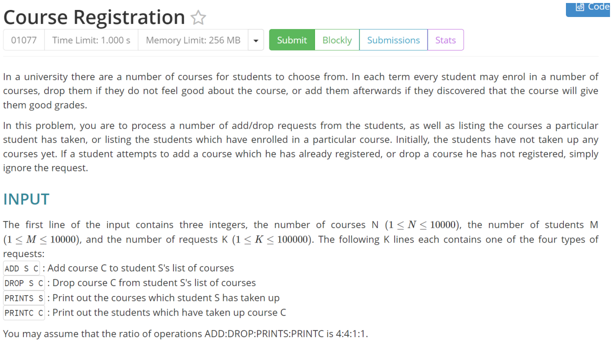
A line containing the word `Yes` if the input string is in parentheses balance or `No` otherwise.

Figure 2: Parenthesis Balance(Source)

Hint

- Stack is the key to this problem
- imagine walking along the string of parenthesis, if you met a close parenthesis
  -)
  -]
  - }
- then what you should you expect when you look back?

Course Registration



Course Registration ☆

01077 | Time Limit: 1.000 s | Memory Limit: 256 MB | Submit | Blockly | Submissions | Stats

In a university there are a number of courses for students to choose from. In each term every student may enrol in a number of courses, drop them if they do not feel good about the course, or add them afterwards if they discovered that the course will give them good grades.

In this problem, you are to process a number of add/drop requests from the students, as well as listing the courses a particular student has taken, or listing the students which have enrolled in a particular course. Initially, the students have not taken up any courses yet. If a student attempts to add a course which he has already registered, or drop a course he has not registered, simply ignore the request.

INPUT

The first line of the input contains three integers, the number of courses N ($1 \leq N \leq 10000$), the number of students M ($1 \leq M \leq 10000$), and the number of requests K ($1 \leq K \leq 100000$). The following K lines each contains one of the four types of requests:
`ADD S C` : Add course C to student S's list of courses
`DROP S C` : Drop course C from student S's list of courses
`PRINTS S` : Print out the courses which student S has taken up
`PRINTC C` : Print out the students which have taken up course C

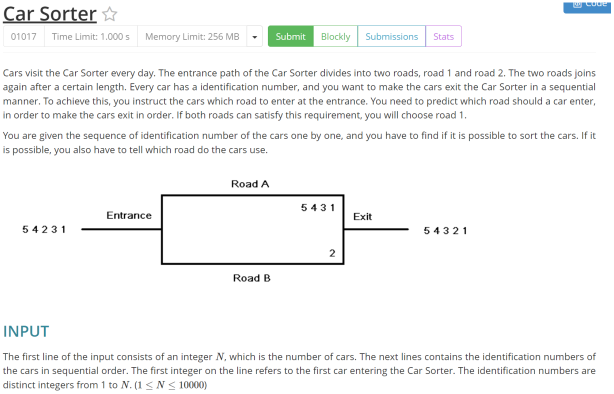
You may assume that the ratio of operations ADD:DROP:PRINTS:PRINTC is 4:4:1:1.

Figure 3: Course Registration(Source)

Hint

- Question suggested the add and drop query is a lot more
- so update should be efficient, but searching need not
- the number of data is fixed (this is important!)
- we don’t need stack or queue here

Car Sorter

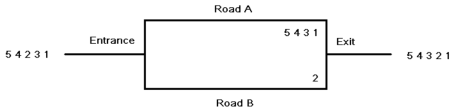


Car Sorter ☆

01017 | Time Limit: 1.000 s | Memory Limit: 256 MB

Cars visit the Car Sorter every day. The entrance path of the Car Sorter divides into two roads, road 1 and road 2. The two roads joins again after a certain length. Every car has a identification number, and you want to make the cars exit the Car Sorter in a sequential manner. To achieve this, you instruct the cars which road to enter at the entrance. You need to predict which road should a car enter, in order to make the cars exit in order. If both roads can satisfy this requirement, you will choose road 1.

You are given the sequence of identification number of the cars one by one, and you have to find if it is possible to sort the cars. If it is possible, you also have to tell which road do the cars use.



INPUT

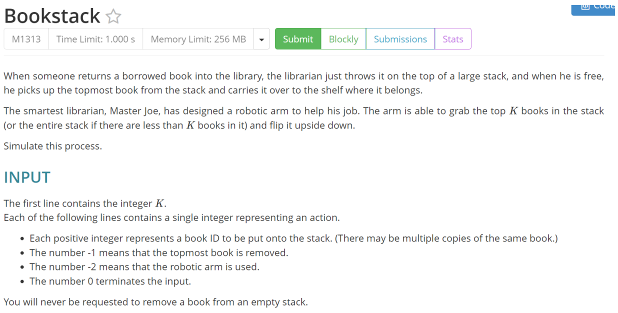
The first line of the input consists of an integer N, which is the number of cars. The next lines contains the identification numbers of the cars in sequential order. The first integer on the line refers to the first car entering the Car Sorter. The identification numbers are distinct integers from 1 to N. $(1 \leq N \leq 10000)$

(Source)

Hint

- consider maintaining two queues
- think carefully which queue you should add the car to

Bookstack


Bookstack ☆

M1313 | Time Limit: 1.000 s | Memory Limit: 256 MB

When someone returns a borrowed book into the library, the librarian just throws it on the top of a large stack, and when he is free, he picks up the topmost book from the stack and carries it over to the shelf where it belongs.

The smartest librarian, Master Joe, has designed a robotic arm to help his job. The arm is able to grab the top K books in the stack (or the entire stack if there are less than K books in it) and flip it upside down.

Simulate this process.

INPUT

The first line contains the integer K.
Each of the following lines contains a single integer representing an action.

- Each positive integer represents a book ID to be put onto the stack. (There may be multiple copies of the same book.)
- The number -1 means that the topmost book is removed.
- The number -2 means that the robotic arm is used.
- The number 0 terminates the input.

You will never be requested to remove a book from an empty stack.

Figure 5: Bookstack(Source)

Hint

- This is a challenge one
- as name suggested you need to use stack
- the number K is important
- a big hint: conasider an array of stack