2015 Fall Computer Science I Program #1: Su-Do-Kode

Please consult Webcourse for the due date/time

Dave has become addicted to Sudoku, the latest puzzle craze in all the newspapers and bookstands. In case you don't know, a Sudoku is a simple number puzzle played on a 3x3 grid of 3x3 subgrids. Below is an example:

	5	7		4	8	9		
			5		9			
	4	8				5	3	6
	2				6			7
	6		1	9	7		8	
7			3				6	
6	3	2				8	5	
			8		3			
		5	2	6		4	7	

Initial Puzzle

3	5	7	6	4	8	9	1	2
2	1	6	5	3	9	7	4	8
9	4	8	7	1	2	5	3	6
5	2	1	4	8	6	3	9	7
4	6	3	1	9	7	2	8	5
7	8	9	3	2	5	1	6	4
6	3	2	9	7	4	8	5	1
1	7	4	8	5	3	6	2	9
8	9	5	2	6	1	4	7	3

Solution

The object of Sudoku is to place numbers 1 through 9 in the empty spaces such that no row, column, or 3x3 subgrid has any number more than once. An interesting property of Sudoku puzzles is that there is always only one possible solution, and it can always be determined using logic, without the need for guessing. Although Dave is wild about Sudoku, he still comes up with incorrect solutions. Dave is tired of being made fun of by his more Sudoku savvy friends, so he's asked you to write a program to check his solutions for him.

Dave would like to give you his completed Sudoku puzzle solutions to have you determine which ones are correct, and which are invalid. For a Sudoku solution to be correct, every row, column, and 3x3 subgrid of the puzzle must have each digit (1 through 9) exactly once.

Input Format (Input will be tested from standard input)

Dave will give you multiple solutions to check. The input will begin with a single integer, *n*, on a line by itself. Following this will be *n* sets of 9 lines, each containing 9 digits. Each of these digits will be in the range 1 through 9, inclusive. Each set of 9 lines of 9 digits represents one of Dave's potential Sudoku puzzle solutions.

Output Specification

For each Sudoku solution, print a single line with either the word `"YES"` or the word `"NO"`.

Sample Input

```
2
357648912
216539748
948712536
521486397
463197285
789325164
632974851
174853629
895261473
263847159
514936278
987125364
645382917
139574826
872619543
658791632
791263485
326458791
```

Sample Output

```
YES
NO
```

Note about reading in a string of digits

There are multiple ways to read a string of digits into an integer array. Here is a segment of code that reads in one string of nine digits and stores it into an integer array of size 9:

```
#define SIZE 9

int i, row[SIZE];
char line[SIZE+1];
scanf("%s", line);

for (i=0; i<SIZE; i++)
    row[i] = line[i] - '0';
```

Since Ascii values of digits are stored contiguously, you can simply subtract the Ascii value of character that is a digit from the character '0' to obtain the value of that digit.

Deliverables

Please submit a single source file, *sudokode.c,* with your solution to this problem via Webcourses before the due date/time for the assignment. Make sure that your program reads from standard in and outputs to standard out, as shown in recitation/lab. It is ***strongly*** suggested that you attempt a submission ***AT LEAST THREE HOURS*** before the actual due date/time to minimize the probability of intervening bad luck. We suggest this precisely because late programs automatically earn a 0 and one can earn up to 50% of the credit for an a program that doesn't work at all. You are allowed to make multiple submissions before the due date of an assignment. (Note: Just for this first assignment, we'll allow a late turn in because of students who may have added the class late. For students who submit after the posted deadline, you will only get one submission, so please keep that in mind.)