

Ship Availability

Objective

Give practice with reading and writing to standard input in C.
Give practice with loops and conditionals in C.
Give practice with strings in C.

Story

Your investors said you couldn't do it, but *buoy* did you show them. You just purchased the Universal Charter Fleet, and it needs a little TLC. All the documents, (e.g. logs, charts, schedules, etc.) were all kept on paper. You have determined to bring the fleet into the modern era by solving some of the problems using your computer science knowledge.

To begin with you can solve the available ship name problem. The Fleet is broken into several Lines. Each Line is composed of several Ships. Each Ship has a name. No you are not naming the Ships (yet).

Currently the Ships of a Line are in some particular use order. Every week the first Ship of the order is used. At the end of the week that Ship is moved to the end of the order, where it will undergo repairs and cleaning during which time the other Ships of the Line will be used. At the end of each week the available Ship of each Line (first in the order) is "rotated" to the end.

Your customers are superstitious and actually care about the names of the ships they cruise on. For this reason your first job will be to report to the customers the names of the ships in use on particular weeks.

Problem

Given the names of the Ships of the Lines, in their operation order, and the weeks customers will come, report to the customers the names of the ships to which they will have access.

Input

The first line of input will contain a single integer, L ($1 \leq L \leq 100$), representing the number of Ship Lines your fleet has. The following L lines will each contain space separated strings of upper and lower case letters, representing a Ship Line description. Each string of letters represents the name of a Ship of the current Line. Ship names are separated by a single space. The Ship names appear in the Ship Line description, in their current use order. The first ship of each line is the ship used in week 1.

Each Ship name will have at least 1 and at most 20 characters. Each Ship Line description will contain at most 100 Ships.

(Input description continued on the next page...)

Following the Ship Line descriptions will be a single line with a single integer, **C** ($1 \leq C \leq 100,000$) representing the number of customers. The following **C** lines will each contain a single integer, **w** ($1 \leq w \leq 100,000$), representing the week in which the customer wishes to cruise.

Output

For each customer output a single line containing the names of the ships that can be used in each Line in the given order of the Lines.

Sample Input	Sample Output
3 KnotLost Fansea Pierless UpDock BoatyMcBoatFace LittleSinker DearlyDeported 5 4 10 1 3 2	UpDock BoatyMcBoatFace DearlyDeported Fanseas BoatyMcBoatFace DearlyDeported KnotLost BoatyMcBoatFace LittleSinker Pierless BoatyMcBoatFace LittleSinker Fanseas BoatyMcBoatFace DearlyDeported
5 A B C D E F G H I J K L M N O P Q 3 20 21 22	E G I K M A F J K N B G H K O

Explanation

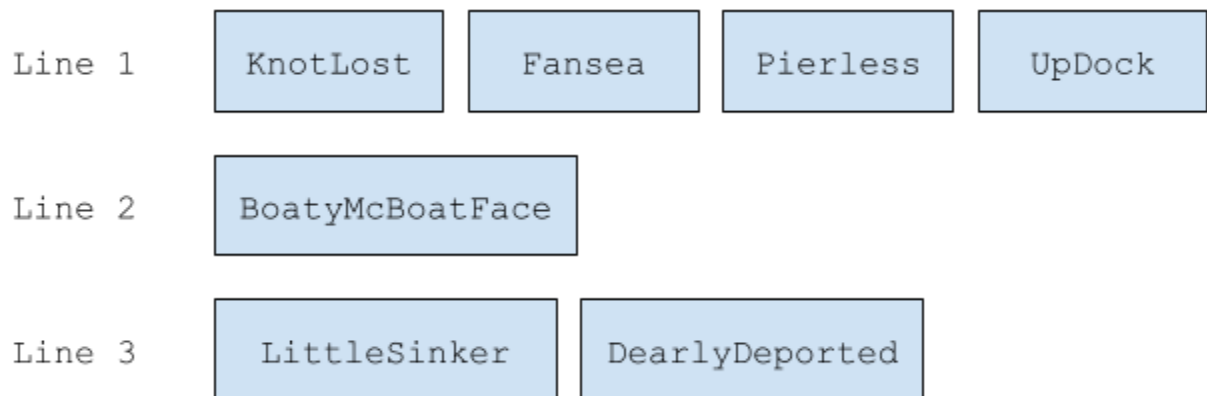
Case 1

There are 3 Lines.

- The first line contains the following 4 boats
 - KnotLost (first up)
 - Fansea
 - Pierless
 - UpDock (last)
- The second line contains the following boat
 - BoatyMcBoatFace
- The third line contains the the following 2 boats
 - LittleSinker
 - DearlyDeported

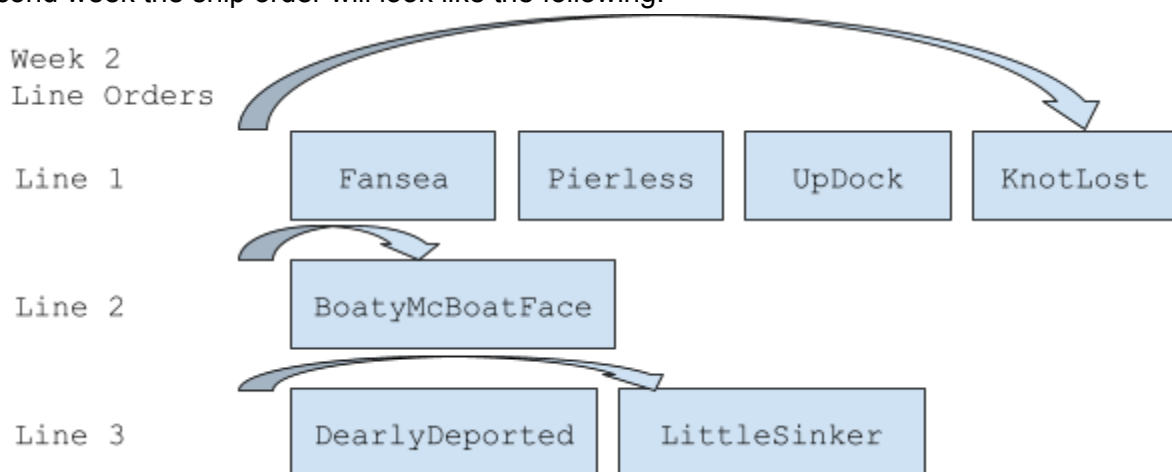
In the first week KnotLost, BoatyMcBoatFace, and LittleSinker will be the used ships. The ship lines look like those of the following image (see next page),

Week 1
Line Orders



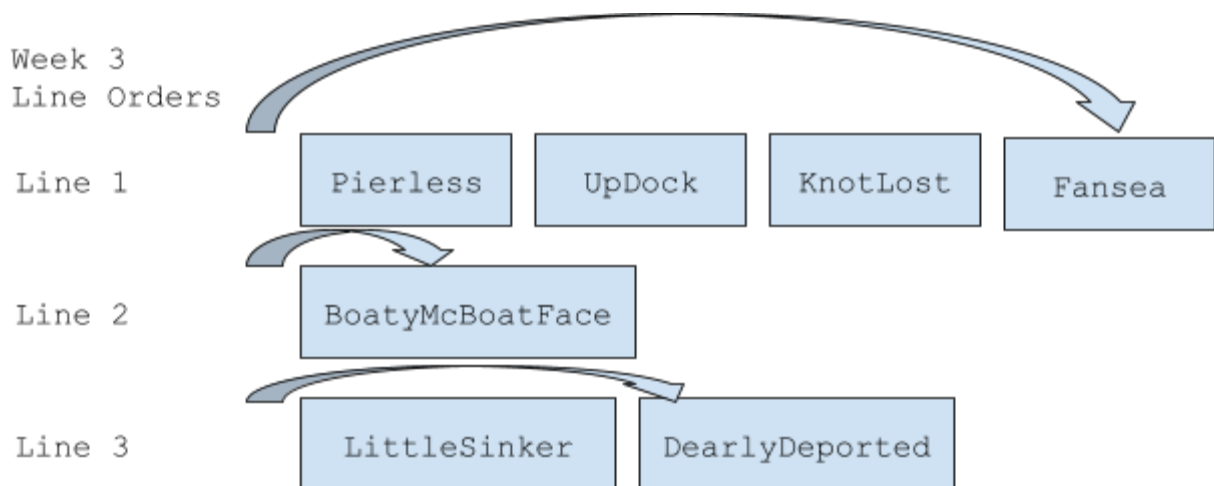
The third customer wanted to know about these boats, so the third line of output contains the first boat of each line.

After the first week is over all three of those boats will move to the end of their order. For the second week the ship order will look like the following.



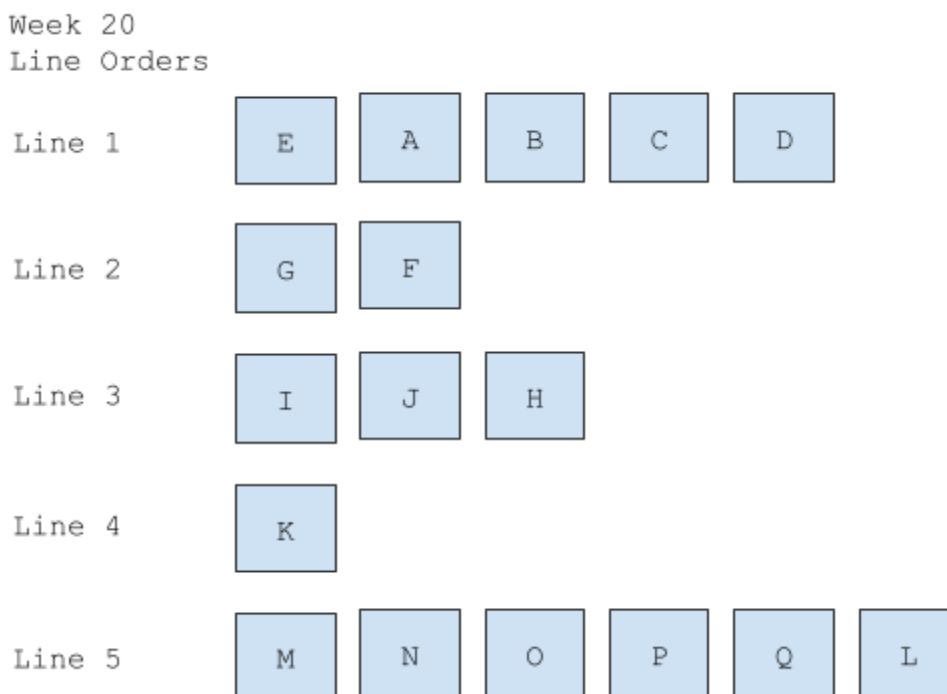
Note that the first ship of each line for the first week moves to the end of the line. In the second week Fansea, BoatyMcBoatFace, and DearlyDeported become the first ship of their respective lines. (Note: that this is the answer for the 5th customer)

The third week (fourth customer's query) has the following ship line status (image on the next page).



Case 2

For the given boat lines, we have the following line orders on week 20.



The ships available during week 20 are E, G, I, K, and M.

Hints

Data Sanitization: In your intro to C class your teacher may have forced you to ensure that the content you are reading is correctly formatted. For this class the input can always assume to be within the input specification. In other words you do NOT need to sanitize the data.

Reading Input: I recommend reading in the first number using traditional scanf. Then use a loop to process each name (maybe with a function). The name processing should involve reading in each character of the name until a new line character is reached, but be careful, since there might be a newline character at the end of the line with the number of names.

Printing Answers: Print the answers as you read in the users. Below is an image of the program running on the first case, where I have typed the input. Note I have put the word output next to the lines that were output from the program. The remaining lines were typed into the program.

	3
	KnotLost Fansea Pierless UpDock
	BoatyMcBoatFace
	LittleSinker DearlyDeported
	5
Output	4
	UpDock BoatyMcBoatFace DearlyDeported
	10
Output	Fanseas BoatyMcBoatFace DearlyDeported
	1
Output	KnotLost BoatyMcBoatFace LittleSinker
	3
Output	Pierless BoatyMcBoatFace LittleSinker
	2
Output	Fanseas BoatyMcBoatFace DearlyDeported

NO QUEUE: You should not use a queue for the assignment...

Modulo: Within each line, the boat order cycles based on the number of ships in the line. For example, suppose the line is "A", "B", and "C". The following lines represent the ship orders for the first few weeks.

"A", "B", "C"	first week
"B", "C", "A"	second week
"C", "A", "B"	third week
"A", "B", "C"	fourth week (repeat of first week)
"B", "C", "A"	fifth week (repeat of second week)
"C", "A", "B"	sixth week (repeat of third week)
"A", "B", "C"	seventh week (repeat of first/fourth week)

The modulo operator (%) can be used to determine the cycle in the array. That is modding by the array length behaves the same as wrapping around the array after moving off the end.

Multidimensional Array: You should consider storing all the boat names in a multidimensional array. Dr. Meade personally used an array with 3 dimensions. The code looked something like the following

```
char names[MAX_LINE_NUMBER][MAX_SHIPS_PER_LINE][MAX_NAME_LENGTH + 1];
```

The array was effectively a 2D array of strings. Since a string is an array of characters, the array is a 3D array.

You will also need an array of integers to hold on to the number of ships in each line.

Dynamic Memory: You don't need dynamic memory...

I BETTER NOT FIND THIS ON THE INTERNET (OUTSIDE OF WEBCOURSES)!!!

Grading Criteria

- Read/Write from/to **standard** input/output (e.g. scanf/printf and no FILE *)
 - 10 points
- Good comments, whitespace, and variable names
 - 15 points
- No extra input output (e.g. input prompts, "Please enter the number of friends:")
 - 10 points
- Read in all the input
 - 5 points
- Loop over all the customers
 - 10 points
- Programs will be tested on 10 cases
 - 5 points each

No points will be awarded to programs that do not compile using "gcc -std=gnu11 -lm".

Sometimes a requested technique will be given, and solutions without the requested technique will have their maximum points total reduced. For this problem use a loop and conditional.
Without this programs will earn at most 50 points!

Any case that causes a program to return a non-zero return code will be treated as wrong. Additionally, any case that takes longer than the maximum allowed time (the max of {5 times my solutions time, 10 seconds}) will also be treated as wrong.

No partial credit will be awarded for an incorrect case.