# 

# 02. Navy Battle

*1914, September 22 – German submarine* [*U-9*](https://en.wikipedia.org/wiki/SM_U-9)[*sinks three unescorted British armored cruisers*](https://en.wikipedia.org/wiki/Action_of_22_September_1914)[*HMS*Aboukir](https://en.wikipedia.org/wiki/HMS_Aboukir_(1900))*,* [*HMS*Hogue](https://en.wikipedia.org/wiki/HMS_Hogue_(1900)) *and* [*HMS*Cressy](https://en.wikipedia.org/wiki/HMS_Cressy_(1899)) *in approximately one hour. Imagine that they had the technology to make themselves a navigational program for the submarine and you are chosen to implement the logic. Navigate U-9 through the battlefield, find and sink the British cruisers in the dark night, avoiding the floating mines all over the North Sea.*

You will be given an integer **n** for the size of the **battlefield** (square shape). On the next **n** lines, you will receive the rows of the **battlefield**. The submarine will start at a **random** position, marked with the letter '**S**'. The submarine surveys the surrounding area through its periscope, so it has to climb up to periscope depth, where it might run across naval **mines**.

When the submarine receives direction, it goes deep and moves **one position toward the given direction**. On each turn, you will be guiding the submarine and giving it the **direction**, in which it should move. The commands will be "**up**", "**down**", "**left**" and "**right**".

When a **new position is reached**, the submarine climbs up to periscope depth to search for a cruiser:

* If a position with '**-**' (dash) is reached, it means that the field is empty and the submarine awaits its next direction.
* **If** it **runs across** a naval **mine (**'**\***'**)**, the submarine takes serious **damage**. When a mine is blown, the position of the mine will be marked with '**-**' (dash). U-9 can **withstand two hits** from naval mines. The **third time** the submarine **is hit** by a mine, it **disappears** and the **mission** **is** **failed. The battle is over** and the following message should be printed on the Console: "**Mission failed, U-9 disappeared! Last known coordinates [{row}, {col}]!**"
* If a battle cruiser is reached **(**'**C**'**)**, the submarine destroys it and the position of the destroyed cruiser will be marked with '**-**' (dash).
* If this is the **last** (third) battle **cruiser** on the **battlefield**, **the battle is over** and the following message should be printed on the Console: "**Mission accomplished, U-9 has destroyed all battle cruisers of the enemy!**"

**The program will end when the battle is over (All battle cruisers are destroyed or the submarine hits mines three times).**

### Input

* On the first line, you are given the integer **n** – the size of the matrix (wall).
* The **next n lines** hold the values for every **row**.
* On each of the next lines, you will get a direction command.

### Output

* If all battle **cruisers** are destroyed, print: "**Mission accomplished, U-9 has destroyed all battle cruisers of the enemy!**"
* If U-9 is hit by a mine three times, print: "**Mission failed, U-9 disappeared! Last known coordinates [{row}, {col}]!**".
* At the end, print the **final state** of the matrix (**battlefield**) with the **last known U-9’s position** on it.

### Constraints

* The size of the **square** matrix (**battlefield**) will be between **[4…10].**
* U-9’s starting position will always be marked with '**S**'.
* There will be always three battle cruisers - fields marked with '**C**'.
* There will be always enough mines on the battlefield to destroy the submarine.
* The commands given will direct the submarine only within the limits of the battlefield.
* You will always receive enough commands to end the battle.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5  \*--\*-  -S-\*C  -\*---  -----  -C-\*C  right  down  left  up  right  right  right  down  down  down  up  left  left  left  down | Mission accomplished, U-9 has destroyed all battle cruisers of the enemy!  \*--\*-  -----  -----  -----  -S-\*- |
| 5  \*--\*-  -S-\*C  -\*---  -----  \*C-\*C  right  right  up  left  left  left | Mission failed, U-9 disappeared! Last known coordinates [0, 0]!  S----  ----C  -\*---  -----  \*C-\*C |