# 2. Rally Racing



*It's time for one of the biggest races in the world, Paris-Dakar. The organizers of the event want you to do a program that helps them track the cars through the separate stages in the event.*

On the first line, you will be given an **integer N**, which represents the **size of a square matrix**. On the second line you will receive the **racing number** of the tracked race car.

On the **next N lines** you will be given the rows of the matrix (**string sequences**, separated by whitespace), whichwill be representing the **race route**. The tracked race **car** **always** starts with **coordinates [0, 0].** Thеre will be a **tunnel** somewhere across the race route. If the race car runs into the tunnel , the **car goes through** it and **exits** at **the other end**. There will be **always two positions** marked with **"T"(tunnel)**. The **finish line** will be marked with **"F".** All other positions will be marked with **"."**.

Keep track of the **kilometers passed**. Every time the race car receives a direction and moves to the **next position** of the race route, it **covers 10 kilometers**. If the car **goes through the tunnel**, it **covers** NOT 10, but **30 kilometers**.

On **each line**, after the matrix is given, you will be receiving **the directions** for the race car.

* **left**
* **right**
* **up**
* **down**

The race car starts moving across the race route:

* If you receive **"End"** command, before the race car manages to reach the finish line, the car is disqualified and the following output should be printed on the Console: **"Racing car {racing number} DNF."**
* If the race car comes across a position marked with **"."**. The car **passes 10 kilometers** for the current move and waits for the next direction.
* If the race car comes across a position marked with **"T"** this is the **tunnel**. The race car goes through it and **moves to the other position marked with "T"** (the other end of the tunnel).The car **passes 30 kilometers** for the current move. The tunnel stays behind the car, so **the race route is clear**, and **both** the positions marked with **"T"**, **should be marked with "."**.
* If the car **reaches the finish line - "F"** position, the race is over. The tracked race car manages to finish the stage and the following output should be printed on the Console: **"Racing car {racing number} finished the stage!"**. Don’t forget that the car has covered another 10 km with the last move.

### Input

* On the first line you will receive N - the size of the square matrix (race route).
* On the second line you will receive the racing number of the tracked car.
* **On the next N lines,** you will receive the **race route** (**elements** will be **separated by a space**).
* On the following lines, you will receive directions **(left, right, up, down).**
* On the last line, you will receive the command **"End"**.

### Output

* If the racing car has reached the finish line before the **"End"** command is given, print on the Console: **"Racing car {racing number} finished the stage!"**
* If the **"End"** command is given and the racing car **has not reached the finish line yet**, the **race ends** and the following message is printed on the Console: **"Racing car {racing number} DNF."**
* On the second line, print the distance that the tracked race car has covered: **"Distance covered {kilometers passed} km."**
* At the end, mark the **last known position of the race car with** **"C" and** print the **final state of the matrix** (race route). If the race car **hasn't gone through the tunnel**, the tunnel exits should be **visualized** in the final state of the matrix. The **row elements** in the output matrix **should NOT be separated** by whitespace.

### Constraints

* The directions will always lead to coordinates in the matrix.
* There will always be two positions marked with **"T"**, representing the tunnel in the race route.
* The size of the **square** matrix (**race route**) will be between **[4…10].**

### Еxamples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comment** |
| 5  01  . . . . .  . . . T .  . . . . .  . T . . .  . . F . .  down  right  right  right  down  right  up  down  right  up  End | Racing car 01 finished the stage!  Distance covered 80 km.  .....  .....  .....  .....  ..C.. | The race car starts moving from position[0,0].  The first command is down, so the moving direction is down. The race car is in position[1,0].  Next three commands are right, so the race car comes across the tunnel – "T". The current car position is [1,3]. Swap the "T" with "." The race car goes through the tunnel, so its next position is [3,1]. Swap the "T" with "."  Next direction is down, so the race car position is [4,1].  Next direction is right, so the race car position is [4,2].  The race car reaches the finish line before the "End" command. So it manages to finish the stage. The remaining directions will be ignored and no more moves are going to be executed. |
| 10  45  . . . . . . . . . .  . . T . . . . . . .  . . . . . . . . . .  . . . . . . . . . .  . . . . . . . . . .  . . . . . . . . . .  . . . . . . F . . .  . . . . . . . . . .  . . . . . . . . . .  . . . . . . . T . .  right  down  down  right  up  left  up  up  End | Racing car 45 DNF.  Distance covered 100 km.  ..........  ..........  ..........  ..........  ..........  ..........  ......F...  ......C...  ..........  .......... |  |