

Czech ACM Student Chapter

Charles University in Prague
Masaryk University
Pavol Jozef Šafárik University in Košice
Slovak University of Technology

Czech Technical University in Prague

Technical University of Ostrava
University of Žilina
Comenius University
University of West Bohemia



CTU Open Contest 2024

Flag Bearer

`bear.(c|cpp|py)`, `Bear.(java|kt)`

The bears have decided to abandon their solitary lives as predators and hunt together. They have formed two groups on the opposite sides of the farm and are preparing to abduct several smaller piglets, whose fate now seems very uncertain.

To prevent their plans from being revealed, the bears will communicate during the operation at a distance using the semaphore alphabet, which they will further encrypt with Caesar's cipher. Caesar's cipher is defined by a single non-negative integer C. Each letter in the message is encrypted by replacing it with the letter that is C positions further in the alphabet. When the position of the used letter goes beyond the end of the alphabet, counting continues cyclically from the start of the alphabet. So, for example, with $C = 5$, the second-to-last letter in the alphabet is encrypted as the fourth letter in the alphabet. Each letter of the semaphore alphabet is encoded by the position of two lower or upper limbs of a signaling mammal that stands upright, facing the recipient of the message. In the diagram shown below, the center of the diagram represents the torso position of the signaling mammal, and its two limbs are represented by two of the eight possible segments radiating from the center. The assignment between limb positions and letters of the alphabet is provided on a separate page (following the samples).

Now, the bears are going to encrypt their short messages. They are good at semaphore alphabet, but they need some help with Caesar's cipher. Please provide them with this assistance.

Input Specification

The first input line contains two integers N ($1 \leq N \leq 26$) and C ($0 \leq C \leq 25$), representing the length of the word to be encrypted and the constant of the Caesar cipher. Next, there are $9N$ lines, each with 9 characters. Each 9 successive lines represent one character in the message. The characters are "`.`", "`*`", or "`#`". Asterisk "`*`" appears only once in each 9 lines and it corresponds to the center of the semaphore scheme. Hash symbols "`#`" correspond to limbs. Coded words use standard English alphabet with 26 characters.

Output Specification

Print the encrypted message, in the same format as the input.

Sample Input 1

```
5 9
.........
.........
.........
.........
.###*....
.....#...
......#..
.......#.
.........
.........
.......#.
......#..
.....#...
....*....
...#.....
..#......
.#.......
.........
.........
.......#.
......#..
.....#...
....*....
....#....
....#....
....#....
.........
.........
.......#.
......#..
.....#...
....*....
....#....
....#....
....#....
.........
.........
....#....
....#....
....#....
.###*....
.........
.........
.........
.........
```

Output for Sample Input 1

```
.........
.........
.........
.........
.###*....
....#....
....#....
....#....
.........
.........
.#.....#.
..#...#..
...#.#...
....*....
.........
.........
.........
.........
.........
.........
.....
....*....
...#.#...
..#...#..
.#.....#.
.........
.........
.........
.........
.........
.....
....*....
...#.#...
..#...#..
.#.....#.
.........
.........
.#.......
..#......
...#.....
....*###.
.........
.........
.........
.........
```

Semaphore alphabet

Down
```
..........
..........
..........
..........
....*.....
....#.....
....#.....
....#.....
..........
```

Left/Down
```
..........
..........
..........
..........
....*.....
...#......
..#.......
.#........
..........
```

Left
```
..........
..........
..........
..........
.###*.....
..........
..........
..........
..........
```

Left/Top
```
..........
.#........
..#.......
...#......
....*.....
..........
..........
..........
..........
```

Top
```
..........
....#.....
....#.....
....#.....
....*.....
..........
..........
..........
..........
```

Top/Right
```
..........
........#.
.......#..
.....#....
....*.....
..........
..........
..........
..........
```

Right
```
..........
..........
..........
..........
....*###..
..........
..........
..........
..........
```

Down/Right
```
..........
..........
..........
..........
....*.....
.....#....
......#...
.......#..
..........
```

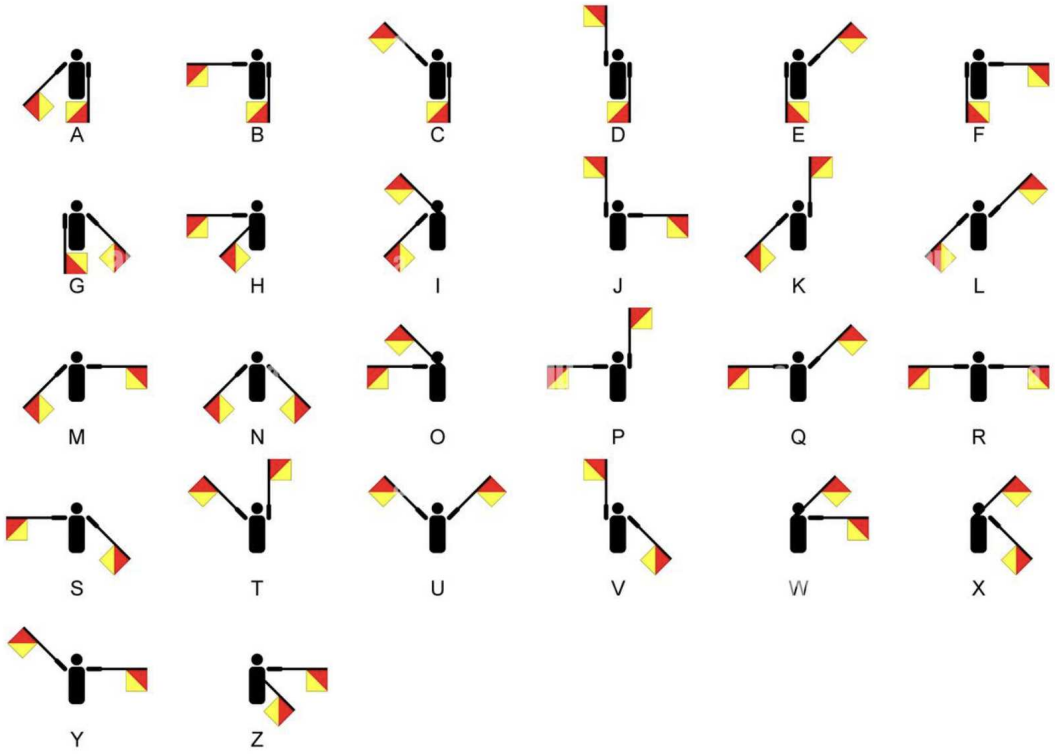


Figure 1: Semaphore Alphabet