## Josephus Permutation

*This problem takes its name by arguably the most important event in the life of the ancient historian Josephus: according to his tale, he and his 40 soldiers were trapped in a cave by the Romans during a siege. Refusing to surrender to the enemy, they instead opted for mass suicide, with a twist: they formed a circle and proceeded to kill one man every three, until one last man was left (and that it was supposed to kill himself to end the act). Well, Josephus and another man were the last two and, as we now know every detail of the story, you may have correctly guessed that they did not exactly follow through the original idea.*

You are now to create a program that prints a **Josephus permutation**, receiving **two lines** of code (the list itself **(string** with elements separated **by a single space)** and a number **k**) as if they were in a circle and **counted out every k** places until none remained.

### Example

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comment** |
| 1 2 3 4 5 6 7  3 | [3,6,2,7,5,1,4] | [1,2,3,4,5,6,7] - initial sequence  [1,2,4,5,6,7] => 3 is counted out and goes into the result [3]  [1,2,4,5,7] => 6 is counted out and goes into the result [3,6]  [1,4,5,7] => 2 is counted out and goes into the result [3,6,2]  [1,4,5] => 7 is counted out and goes into the result [3,6,2,7]  [1,4] => 5 is counted out and goes into the result [3,6,2,7,5]  [4] => 1 is counted out and goes into the result [3,6,2,7,5,1]  [] => 4 is counted out and goes into the result [3,6,2,7,5,1,4] |