List Replication

Given a list, repeat each element in the list n amount of times. The input and output portions will be handled automatically by the grader. You need to write a function with the recommended method signature.

Input Format

The first line contains the integer S where S is the number of times you need to repeat the elements. The next X lines each contain an integer. These are the X elements in the array.

Output Format

Output each element of the original list S times, each on a separate line. You have to return the list/vector/array of $S * X$ integers. The relative positions of the values should be the same as the original list provided in the input.

Constraints

$0 <= X <= 10$
$1 <= S <= 100$

Sample Input

```
3
1
2
3
4
```

Sample Output

```
1
1
1
2
2
2
3
3
3
4
4
4
```

Recommended Method Signature

```
Number Of Parameters: 2
Parameters: [number of times to replicate elements, list]
Returns: List or Vector
```

For Hackers Using Clojure

This will be the outline of your function body (fill in the blank portion marked by underscores):

```
(fn[num lst]___________________________)
```

For Hackers Using Scala

This will be the outline of your function body (fill in the blank portion marked by underscores):

```
def f(num:Int,arr:List[Int]):List[Int] = __________________
```

For Hackers Using Haskell

This will be the outline of your function body (fill in the blank portion marked by underscores):

```
f n arr = ___________________
```

For Hackers Using OCaml

This will be the outline of your function body (fill in the blank portion marked by underscores):

```
let f n arr = (*Complete this function*)
```

For Hackers Using other Languages

You have to read input from standard input and write output to standard output. Please follow the input/output format mentioned above.

NOTE: You only need to submit the code above after filling in the blanks appropriately. The input and output section will be handled by us. The focus is on writing the correct function.