

Pile of books

Input file: **standard input**
Output file: **standard output**
Time limit: 0.5 seconds
Memory limit: 64 megabytes

Davide purchased N books in preparation for the new academic year. He carefully placed them on a table in his study room.

After a few minutes, he noticed that the books were arranged in ascending order, with the smallest book at the bottom and the largest book at the top.

Realizing this, Davide decided that he preferred the books to be in a different order.

He wants to reverse their position, so the largest book is at the bottom and the smallest book is at the top.

Davide can perform the following operation a maximum of T (10000) times: he can remove K books from the top of pile A and place them, in the same order, on top of pile B ($0 \leq A, B \leq 10^9$) (he may choose to utilize a new empty pile, if desired).

The cost of the operation will be 1 if the movement of books results in pile A having less books than pile B prior to the movement, and 0 otherwise.

Help Davide to reverse the order of the books while minimizing the total cost.

Input

The first line contains the only integer N ($1 \leq N \leq 1000$), the number of books.

Your program will be tested against several test cases grouped in subtasks. The score in each subtask will be calculated as the minimum score obtained in any of its test cases, multiplied by the value of the subtask.

The score of a test case is 0 if the answer is wrong or invalid, otherwise let C be the total cost of the operations performed by your program. It can be proved that with the given constraints the minimal cost is at least 1, and a solution always exists with at most T operations. Your score will be calculated based on the following formula:

score =

$$\min\left(1 - \frac{\log_2(C) - 1}{\log_2(T) - 1}, 1\right)$$

.

Output

For each operation, output a line containing three integers K , A and B .

Examples

standard input	standard output
3	1 0 1 1 0 2 1 0 3 1 1 0 1 2 0 1 3 0
4	1 0 1 1 0 1 1 0 2 1 0 2 2 2 1

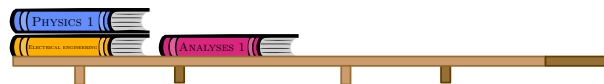
Note

In the **first sample case** initially there are 3 books on pile 0.

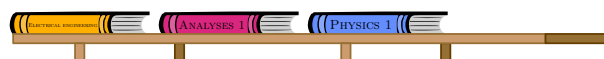


We perform the following operations:

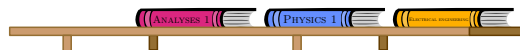
- Move 1 book from pile 0 to pile 1 with cost 0.



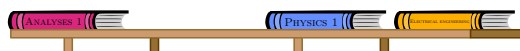
- Move 1 book from pile 0 to pile 2 with cost 0.



- Move 1 book from pile 0 to pile 3 with cost 0.



- Move 1 book from pile 1 to pile 0 with cost 0.



- Move 1 book from pile 2 to pile 0 with cost 1.



- Move 1 book from pile 3 to pile 0 with cost 1.



Finally, we have 3 books on pile 0 in the correct order, with a total cost of 2.

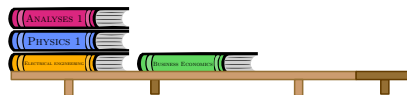
Note that for 3 books there exists a solution with cost 1, too, however according to the scoring formula, the solution with cost 2 gets a full score as well.

In the **second sample case** initially there are 4 books on pile 0.



We perform the following operations:

- Move 1 book from pile 0 to pile 1 with cost 0.



- Move 1 book from pile 0 to pile 1 with cost 0.



- Move 1 book from pile 0 to pile 2 with cost 0.



- Move 1 book from pile 0 to pile 2 with cost 1.



- Move 2 books from pile 2 to pile 1 with cost 1.



Finally, we have 4 books on pile 1 in the correct order, with a total cost of 2.