Emoji di compleanno (compleanno)

Today, not unlike most days, William woke up to a Facebook notification reminding him of one of his friends' birthday! He is a big fan of birthdays. To celebrate, he wants to post a message to his friend's timeline, along with a funny GIF and a bunch of tiny birthday cake emojis... Just like this one: 🎂

In order to differentiate his post from the countless others, William is used to adding a number of emojis that is equal to the new age of his friend.

For instance, if said friend was turning 18 today, the text of the message would be:

"🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂"

Due to his logical nature, William wishes to write these 18 emojis in the most efficient way possible. Indeed, he has several choices for composing the message: a first (pretty naïve) way consists in typing the 18 emojis one at a time; on the other hand, a more efficient way relies on a clever use of the "copy-paste" function. Specifically, he has 3 operations available:

- Insert a 🎂 emoji from the list of emojis provided by Facebook. This operation requires **1 click**.
- Use the mouse to select all the text, right-click and choose "Copy", thus saving all the emojis written so far in the clipboard. This operation requires **2 clicks**.
- Right-click using the mouse and select "Paste", thus appending the emojis stored in the clipboard at the end of the message. This operation requires **1 click**.

We say that a choice of operations for writing the emojis is **optimal** if it requires the least possible number of clicks. Since lately William has been receiving a lot of Facebook birthday notifications every day (thanks to all those friendship requests from kitten-related groups), he decided to write a program that finds the optimal strategy for him. Help him write the program!

Note that more than one optimal strategy can exist, that is: there might be many different ways of typing the emojis with the minimum number of operations. It suffices that your program implements any one such strategy.

Implementation

You should submit a single file, with either a `.c` or `.cpp` extension.

> ☞ Among the attachments in this task you will find a template `compleanno.c` or `compleanno.cpp` with a sample implementation.

You have to implement the following function:

C/C++	`void auguri(long long int N);`

- The integer N represents the age of your Facebook friend on his birthday.

Your program can interact with the following functions, defined in the grader:

C/C++	`void aggiungi();`
	`void copia();`
	`void incolla();`

- Calling `aggiungi()` is equivalent to 1 click on the 🎂 emoji.
- Calling `copia()` is equivalent to 2 clicks: one to select all, and one to copy the text.
- Calling `incolla()` is equivalent to 1 click to paste the text.

The grader will call the function `auguri(N)`, with the age as parameter. Your function can call functions `aggiungi()`, `copia()`, and `incolla()`. At the end of execution, the grader will record the total number of clicks made in the output file.

Sample grader

Among this task's attachments you will find a simplified version of the grader used during evaluation, which you can use to test your solutions locally. The sample grader reads data from `stdin`, calls the functions that you should implement and writes back on `stdout` using the following format.

The input file is made up of a single line containing the integer N, the age of your friend.

The output file is made up of a single line containing one integer, the total number of clicks.

Constraints

- $1 \leq N \leq 10^{18}$.

Scoring

Your program will be tested on a number of testcases grouped in subtasks. In order to obtain the score associated to a subtask, you need to correctly solve all the testcases it contains.

- **Subtask 1 [0 points]**: Sample test cases.
- **Subtask 2 [7 points]**: N is a power of 2, that is: $N = 2^k$ for some k.
- **Subtask 3 [6 points]**: $N \leq 50$.
- **Subtask 4 [5 points]**: $N \leq 800$.
- **Subtask 5 [19 points]**: $N \leq 10000$.
- **Subtask 6 [17 points]**: $N \leq 500000$.
- **Subtask 7 [22 points]**: $N \leq 10^9$.
- **Subtask 8 [8 points]**: $N \leq 10^{12}$.
- **Subtask 9 [15 points]**: $N \leq 10^{15}$.
- **Subtask 10[1 point]**: No additional constraint.

Examples

stdin	stdout
8	7
18	10

Explanation

For the **first sample case**, a possible strategy is the following:

1. Click the emoji. The text becomes: 🎂
2. Click the emoji again. The text becomes: 🎂🎂
3. Click the emoji again. The text becomes: 🎂🎂🎂
4. Click the emoji again. The text becomes: 🎂🎂🎂🎂
5. Use your mouse to select everything you wrote.
6. Right-click and select “Copy”.
7. Right-click and select “Paste”. **Done!**

For the **second sample case**, a possible strategy is the following:

1. Click the emoji. The text becomes: 🎂
2. Click the emoji again. The text becomes: 🎂🎂
3. Click the emoji again. The text becomes: 🎂🎂🎂
4. Use your mouse to select everything you wrote.
5. Right-click and select “Copy”.
6. Right-click and select “Paste”. The text becomes: 🎂🎂🎂🎂🎂🎂
7. Right-click and select “Paste” again. The text becomes: 🎂🎂🎂🎂🎂🎂🎂🎂🎂🎂
8. Use your mouse to select everything you wrote.
9. Right-click and select “Copy”.
10. Right-click and select “Paste”. **Done!**