

A. Plus and Square Root

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

ZS the Coder is playing a game. There is a number displayed on the screen and there are two buttons, '+' (plus) and '√' (square root). Initially, the number 2 is displayed on the screen. There are $n + 1$ levels in the game and ZS the Coder start at the level 1.

When ZS the Coder is at level k , he can :

1. *Press the '+' button.* This increases the number on the screen by exactly k . So, if the number on the screen was x , it becomes $x + k$.
2. *Press the '√' button.* Let the number on the screen be x . After pressing this button, the number becomes \sqrt{x} . After that, ZS the Coder levels up, so his current level becomes $k + 1$. This button can only be pressed when x is a **perfect square**, i.e. $x = m^2$ for some positive integer m .

Additionally, after each move, if ZS the Coder is at level k , and the number on the screen is m , then **m must be a multiple of k** . Note that this condition is only checked after performing the press. For example, if ZS the Coder is at level 4 and current number is 100, he presses the '√' button and the number turns into 10. Note that at this moment, 10 is not divisible by 4, but this press is still valid, because after it, ZS the Coder is at level 5, and 10 is divisible by 5.

ZS the Coder needs your help in beating the game — he wants to reach level $n + 1$. In other words, he needs to press the '√' button n times. Help him determine the number of times he should press the '+' button before pressing the '√' button at each level.

Please note that ZS the Coder wants to find just any sequence of presses allowing him to reach level $n + 1$, but not necessarily a sequence minimizing the number of presses.

Input

The first and only line of the input contains a single integer n ($1 \leq n \leq 100\,000$), denoting that ZS the Coder wants to reach level $n + 1$.

Output

Print n non-negative integers, one per line. i -th of them should be equal to the number of times that ZS the Coder needs to press the '+' button before pressing the '√' button at level i .

Each number in the output should not exceed 10^{18} . However, the number on the screen **can be greater** than 10^{18} .

It is guaranteed that at least one solution exists. If there are multiple solutions, print any of them.

Examples

input
3
output
14
16
46
input
2

Codeforces Round #372 (Div. 1)

Finished

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest



→ Problem tags

constructive algorithms

math

No tag edit access

→ Contest materials

- Announcement 
- Tutorial 

output
9999999999999999998 44500000000

input
4
output
2 17 46 97

Note

In the first sample case:

On the first level, ZS the Coder pressed the '+' button 14 times (and the number on screen is initially 2), so the number became $2 + 14 \cdot 1 = 16$. Then, ZS the Coder pressed the " button, and the number became .

After that, on the second level, ZS pressed the '+' button 16 times, so the number becomes $4 + 16 \cdot 2 = 36$. Then, ZS pressed the " button, levelling up and changing the number into .

After that, on the third level, ZS pressed the '+' button 46 times, so the number becomes $6 + 46 \cdot 3 = 144$. Then, ZS pressed the " button, levelling up and changing the number into .

Note that 12 is indeed divisible by 4, so ZS the Coder can reach level 4.

Also, note that pressing the '+' button 10 times on the third level before levelling up does not work, because the number becomes $6 + 10 \cdot 3 = 36$, and when the " button is pressed, the number becomes 6 and ZS the Coder is at Level 4. However, 6 is not divisible by 4 now, so this is **not a valid solution**.

In the second sample case:

On the first level, ZS the Coder pressed the '+' button 9999999999999999998 times (and the number on screen is initially 2), so the number became $2 + 9999999999999999998 \cdot 1 = 10^{18}$. Then, ZS the Coder pressed the " button, and the number became .

After that, on the second level, ZS pressed the '+' button 44500000000 times, so the number becomes $10^9 + 44500000000 \cdot 2 = 9 \cdot 10^{10}$. Then, ZS pressed the " button, levelling up and changing the number into .

Note that 300000 is a multiple of 3, so ZS the Coder can reach level 3.