Task: KWA
Squares

Olimpiada Informatyczna

XXII OI, Stage I. Source file `kwa.*` Available memory: 64 MB. *6.10–3.11.2014*

We consider decompositions of positive integers into sums of *unique* squares of positive integers, called decompositions in short from now on. For example, the number 30 has two decompositions: $1^2 + 2^2 + 5^2 = 1^2 + 2^2 + 3^2 + 4^2 = 30$, whereas the number 8 has none.

Specifically, we are interested in how large the largest square in the decomposition of a given number n has to be. In other words, we want to determine the value $k(n)$, defined as the minimum over decompositions of n of the maximum integer (not its square!) in the decomposition. We assume that $k(n) = \infty$ if n cannot be decomposed. For example, $k(1) = 1$, $k(8) = \infty$, $k(30) = 4$, $k(378) = 12$, $k(380) = 10$.

We call an integer x *overgrown* if there is an integer $y > x$ such that $k(y) < k(x)$. It follows from the previous example that 378 is overgrown.

For a given integer n, you are to determine $k(n)$ and the number of overgrown integers no larger than n.

Input

In the first and only line of the standard input, there is a single integer n ($1 \leq n \leq 10^{18}$). There is a set of tests worth 45% of total score, in which $n \leq 50\,000\,000$ holds, a subset of these worth 30% of total score, in which $n \leq 1\,000\,000$ holds, and an even smaller subset of the latter worth 20% of total score, in which $n \leq 1000$ holds.

Output

Your program should print two integers, separated by a single space, to the standard output: first $k(n)$ and then the number of overgrown integers in the range from 1 to n. If $k(n) = \infty$, then `-` (dash or minus sign) should be printed instead of the first number.

Example

For the input data:

```
30
```

the correct result is:

```
4 15
```

whereas for the following input data:

```
8
```

the correct result is:

```
- 5
```