University of Cape Town ~ Department of Computer Science
Computer Science 3003S Theory of Algorithms ~ 2014

Practical Test 2 Session 1 Question 1

Time: 120 minutes

Problem Description

A collection of precious gems is being sold. In an attempt to prevent a single buyer from purchasing them all, the cost increases, greater than linearly, with the number of gems bought.

The total cost to buy K gems is f(K) where f(K) is defined as the sum of j * (K/j) for all integers 0 < j < K. *(Note that K/j here uses integer division, so the result has no fractional part. For example: 8/3 would equal 2.)*

As an example, to buy 5 gems, the cost, f(5), would be calculated as:

1 * (5/1) + 2 * (5/2) + 3 * (5/3) + 4 * (5/4) = 1*5 + 2*2 + 3*1 + 4*1 = 16

Write a program that, given an amount of currency, N, calculates the greatest value of K for which f(K) ≤ N i.e. Write a program that calculates the largest number of gems that can be bought for a given amount of money.

Constraints:

1 ≤ K ≤ 1,000,000
This means that the input value N is bounded by:
1 ≤ N ≤ f(1,000,000) i.e. 1 ≤ N ≤ 822467118437

Example:

Given N = 30:

f(6) = 1 * (6/1) + 2 * (6/2) + 3 * (6/3) + 4 * (6/4) + 5 * (6/5)
= 1*6 + 2*3 + 3*2 + 4*1 + 5*1
= 27

f(7) = 1 * (7/1) + 2 * (7/2) + 3 * (7/3) + 4 * (7/4) + 5 * (7/5) + 6 * (7/6)
= 1*7 + 2*3 + 3*2 + 4*1 + 5*1 + 6*1
= 34

So the maximum number of gems that could be purchased would be 6.

Note that the function f(K) is *strictly increasing*.

Note that values used in this question can be larger than the maximum value of a 32 bit integer type, requiring the use of 64 bit integer types (long in Java, long long in C and C++).

Question continues on next page

File names

- Use `pricing.c` if you are writing your program in C.
- Use `pricing.cpp` if you are writing your program in C++.
- Use `Pricing.java` if you are writing your program in Java.

Please remember to zip your file.

Input and Output

Program input and output will make use of `stdio` streams (`System.in` and `System.out` in Java) i.e. not file I/O.

Input consists of a single line containing a single integer value, N, the maximum amount of currency that can be spent.

Output consists of a single integer, K (the maximum number of gems that can be purchased), followed by a line break --- in Java, for example, use `System.out.println`, not `System.out.print`. The automatic marker expects output in this precise form.

Sample Input:

```
30
```

Sample output:

```
6
```

Scoring

Each test case that is answered correctly will earn 10 points.

END