

Assignment 5

CS329e - Elements of Software Design
Working Hard Problem
(100 points)

Due Date on Canvas and Gradescope

1 Description

Chris has to complete a programming assignment overnight. He has to write n lines of code before morning. He is dead tired and he tries drinking some black coffee to keep him awake. But each time he drinks a cup of coffee he stays awake for a short amount of time but his productivity goes down by a constant factor k

This is how he plans to write the program. He will write the first v lines of code, then drink his first cup of coffee.

- Since his productivity has gone down by a factor of k he will write $v // k$ lines of code.
- He will have another cup of coffee and then write $v // k^2$ lines of code.
- He will have another cup of coffee and write $v // k^3$ lines of code and so on.
- He will collapse and fall asleep when $v // k^p$ becomes 0.

Now Chris does want to complete his assignment and maximize on his sleep. So he wants to figure out the minimum allowable value of v for a given productivity factor that will allow him to write at least n lines of code before he falls asleep.

Input:

You will read your input from standard input as given in the following format file [work.in](#):

```
1 2
2 300 2
3 59 9
```

The first line is T the number of test cases. This will be followed by T lines of input. Each line of input will have two numbers n and k . n is the number of lines of code to write and k is the productivity factor, where $1 \leq n \leq 10^6$ and $2 \leq k \leq 10$.

Output:

For each test case write your result to standard out as shown in file [work.out](#). In your output there will be v lines of code the Chris has to write, as well as the time it took for each function. For the above two test cases, the output will be:

```
1 Binary Search: 152
2 Time: 9.512901306152344e-05
3
4 Linear Search: 152
5 Time: 0.0005910396575927734
6
7
8 Binary Search: 54
9 Time: 4.696846008300781e-05
10
11 Linear Search: 54
12 Time: 9.01222290039062e-05
13
```

Do not worry if your times do not match ours exactly. They are given just for comparison purposes. For this assignment, main has been written completely for you, and nothing needs to be changed in it.

You will be solving this problem in 2 ways. First, you will write a function that uses a linear search to solve the problem. Then you will write a function that uses a modified binary search algorithm to solve it again. Both functions will return the same answer, but the binary search method will usually be faster.

It is recommended that you write a helper function, which given a value v representing the number of lines Chris writes before his first cup of coffee and a value k , the productivity factor, will calculate the number of lines Chris will write before falling asleep. This can be called in both the linear and binary functions to make the computations easier.

For this assignment you may work with a partner. Both of you must read the paper on Pair Programming¹ and abide by the ground rules as stated in that paper. If you are working with a partner then only one of you will be submitting the code. But make sure that your partner's name and UT EID is in the header. If you are working alone then remove the partner's name and eid from the header.

1.1 Turnin

Turn in your assignment on time on Gradescope system on Canvas. For the due date of the assignments, please see the Gradescope and Canvas systems.

1.2 Academic Misconduct Regarding Programming

In a programming class like our class, there is sometimes a very fine line between "cheating" and acceptable and beneficial interaction between students (In different assignment groups). Thus, it is very important that you fully understand what is and what is not allowed in terms of collaboration with your classmates. We want to be 100% precise, so that there can be no confusion.

The rule on collaboration and communication with your classmates is very simple: you cannot transmit or receive code from or to anyone in the class in any way – visually (by showing someone your code), electronically (by emailing, posting, or otherwise sending someone your code), verbally (by reading code to someone) or in any other way we have not yet imagined. Any other collaboration is acceptable.

The rule on collaboration and communication with people who are not your classmates (or your TAs or instructor) is also very simple: it is not allowed in any way, period. This disallows (for example) posting any questions of any nature to programming forums such as **StackOverflow**. As far as going to the web and using Google, we will apply the "**two line rule**". Go to any web page you like and do any search that you like. But you cannot take more than two lines of code from an external resource and actually include it in your assignment in any form. Note that changing variable names or otherwise transforming or obfuscating

¹Read this paper about Pair Programming <https://collaboration.csc.ncsu.edu/laurie/Papers/Kindergarten.PDF>

code you found on the web does not render the "two line rule" inapplicable. It is still a violation to obtain more than two lines of code from an external resource and turn it in, whatever you do to those two lines after you first obtain them.

Furthermore, you should cite your sources. Add a comment to your code that includes the URL(s) that you consulted when constructing your solution. This turns out to be very helpful when you're looking at something you wrote a while ago and you need to remind yourself what you were thinking.

We will use the following Code plagiarism Detection Software to automatically detect plagiarism.

- **Stanford MOSS**

`https://theory.stanford.edu/~aiken/moss/`

- **Jplag - Detecting Software Plagiarism**

`https://github.com/jplag/jplag and https://jplag.ipd.kit.edu/`