

Programming Assignment 1 – CMSC351 Fall 2017

Professor: M.T. Hajiaghayi ¹

¹Department of Computer Science, University of Maryland

- You can collaborate with other students on the solutions but do not share your codes with each other.
- You can only use Java 1.8.
- You will have to submit only one file, named *Main.java*, to the submit server.
- More details on the formatting of the assignment will be posted on piazza. Make sure to follow them as the codes will be automatically graded.
- Check the [course website](#) for the due date.

1 The Sorting Algorithm

Soben, a young algorithm designer, has come up with a sorting algorithm that takes exactly $\lfloor a \cdot n \ln(n) + b \rfloor$ nanoseconds to sort an array of n numbers (a and b are constant numbers and the logarithm is natural, i.e., it is in base e). He is now interested to find the size of the largest array that can be sorted using his algorithm in t nanoseconds. More precisely, for a given time t nanoseconds, Soben wants to find the maximum number n such that $\lfloor a \cdot n \log(n) + b \rfloor \leq t$. Help him find this number.

Input

- The first line of input contains constant a ($1 \leq a \leq 10$).
- The second line of input contains constant b ($0 \leq b \leq 10^6$).
- The third line of each scenario, contains the time t ($1 \leq t \leq 10^{16}$).

Output

- Print the maximum number of elements that can be sorted in time t .

Sample

Sample input	Sample output
1 0 1000	190
10 55 20008973878324	79711685664

2 Non-overlapping Intervals

As an undergraduate student, Soben performed so well in CMSC351 that he is now a TA of this course. His office hours are usually so crowded that he is not able to answer all the questions. To maximize the number of students who receive their answers, Soben asks each student to write down one interval on the whiteboard: the time range that they need to talk to him. He then chooses the maximum number of *non-overlapping intervals* (two intervals $[a, b]$ and $[c, d]$ overlap if and only if $(c \leq a \leq d) \vee (a \leq c \leq b)$) and answers the corresponding students. Unfortunately, the process of choosing the maximum number of non-overlapping intervals is time consuming and wastes a large fraction of his office hour. Help him write a program that automates this.

Input

- The first line of the input contains n , the number of intervals ($n \leq 10^6$). It is then followed by n lines, each specifying one interval.
- For $1 \leq i \leq n$, line $i + 1$ of the input describes the i -th interval. Each interval is described by two positive integers a_i and b_i separated by a space ($1 \leq a_i \leq b_i \leq 10^9$).

Output

- Print the maximum number of non-overlapping intervals.

Sample

Sample input	Sample output
3	2
1 10	
10 20	
20 30	

Sample input	Sample output
5	3
8 10	
1 5	
2 7	
7 9	
10 12	