Indian Institute of technology, Guwahati
Department of Computer Science and Engineering
Data Structure Lab (CS210) Assignment: 5

Date: 27th August 2018. **Total Marks: 20**

1) A teacher has given a string to all the students in the class and said rearrange the string as you like. Bob is one of the student in the class and he wants to be intelligent in the class, he wants to rearrange the string in a very smart way. He wants to rearrange the characters in the string so that no two adjacent characters are same. Bob facing difficulty while doing that, so he decided to write a programme, but Bob is not good at programming, please help Bob. **(10)**

   Note

   string contains only lower case English alphabets, and don't use any sorting technique.

   Hint

   Use augmented priority queue.

   Input Format

   First line of input contains a string

   Output Format

   If it is possible then, print the string after rearrangement (if many such arrangements possible print any one of them).

   If it is not possible then, print "Not Possible" (without quotes)

   Sample Input 1

   aaabc

   Sample Output 1

   abaca

   Sample Input 2

   aaaabc

   Sample Ouput 2

   Not Possible

2) Bob owns a cake restaurant and he manages it in his own way. While in a normal restaurant, a customer is served by following the first-come, first-served rule, Bob simply minimizes the average waiting time of his customers. So he gets to decide who is served first, regardless of how sooner or later a person comes.

   Different kinds of cakes take different amounts of time to cook. Also, once he starts cooking a cake, he cannot cook another cake until the first cake is completely cooked. Let's say we have three customers who come at time t=0, t=1, & t=2 respectively, and the time needed to cook their cakes is 3, 9, & 6 respectively. If Bob applies first-come, first-served rule, then the waiting time of three customers is 3, 11, & 16 respectively. The average waiting time in this case is (3 + 11 + 16) / 3 = 10. This is not an optimized solution. After serving the first customer at time t=3, Bob can choose to serve the third customer. In that case, the waiting time will be 3, 7, & 17 respectively. Hence the average waiting time is (3 + 7 + 17) / 3 = 9.

   Help Bob achieve the minimum average waiting time. For the sake of simplicity, just find the integer part of the minimum average waiting time. **(10)**

   Input Format

   - The first line contains an integer N, which is the number of customers.

- In the next N lines, the i^{th} line contains two space separated numbers T_i and L_i. T_i is the time when i^{th} customer order a cake, and L_i is the time required to cook that cake.
- The customer is not the customer arriving at the arrival time.

Output Format

- Display the integer part of the minimum average waiting time.

Constraints

- $1 \leq N \leq 100$
- $0 \leq T_i \leq 10^9$
- $1 \leq L_i \leq 10^9$

Note

- The waiting time is calculated as the difference between the time a customer orders pizza (the time at which they enter the shop) and the time she is served.
- Cook does not know about the future orders.

Sample Input 1

3
0 3
1 9
2 6

Sample Output 1

9

Explanation 1

Serving 1^{st} customer first, so waiting time is 3
Serving 3^{rd} customer second, so waiting time is 7
Serving 2^{nd} customer third, so waiting time is 17
Average waiting time is: (3+7+17)/3 = 9

Sample Input 2

3
0 9
10 3
10 4

Sample Output 2

6

Explanation 2

Serving 1^{st} customer first, so waiting time is 9
Serving 2^{nd} customer second, so waiting time is 3
Serving 3^{rd} customer third, so waiting time is 7
Average waiting time is (9+3+7)/3 = 6.66, but integral part is 6.

Evaluation Guidelines:

1. Full marks if both methods work for all test cases.
2. If code is not working but most of code is written, maximum 40% can be given based on TA's evaluation.
3. 15% marks will be deducted for each test not running.
4. At most 10% marks will be deducted for bad coding style. i.e., (1) code is not modular (2) code is not properly indented (3) code is not properly commented and (4) Variable and functions are not suitably named.