
CS29003 Algorithms Laboratory
Class Test 1
Date: February 23, 2021

General instruction to be followed strictly

1. Do not use any global variable unless you are explicitly instructed so.
2. Do not use Standard Template Library (STL) of C++.
3. Use proper indentation in your code and comment.
4. Name your file as <roll_no>_<assignment_no>. For example, if your roll number is 14CS10001 and you are submitting assignment 3, then name your file as 14CS10001_3.c or 14CS10001_3.cpp as applicable.
5. Write your name, roll number, and assignment number at the beginning of your program.
6. Make your program as efficient as possible.

There are n restaurants in your city, namely R_1, \dots, R_n . Your friend has come to your home for a vacation. You want to take your friend for lunch to each of these n restaurants on n different days. Your friend gets excited to hear your proposal and quickly plans these outings. An outing plan specifies the day to visit each restaurant. For $n = 3$, the following is an example of a plan

3, 2, 7

which reads as visit R_1 on day 3, R_2 on day 2, and R_3 on day 7. Visiting each restaurant costs you 1000 rupees. If you go to a restaurant some day, you will not be able to go to your work that day. You go to work everyday except those days when you go to some restaurant with your friend. In any day, if you go to work, then you earn 1000 rupees at the end of the day. On day 0, you do not have any money with you. You also want to spend any amount of money only if you have at least that much with you at that point of time. This makes some plans for visiting restaurants infeasible. For example, the plan 3, 2, 7 (for $n = 3$) is an infeasible plan since at the end of day 2, you do not have any money left and thus you cannot visit any restaurant (R_1 in particular) on day 3. Hence, your goal is to compute a feasible plan which differs from the proposed plan as less as possible. The difference between two plans x_1, x_2, \dots, x_n and y_1, y_2, \dots, y_n is given by

$$\sum_{i=1}^n |x_i - y_i|$$

For example, the closest feasible plan from 3, 2, 7 is 4, 2, 7 which differs by 1 from the plan 3, 2, 7.

Take the number n of restaurants and a plan x_1, x_2, \dots, x_n as input from the user. Store the input data in a data structure which you find appropriate. Now design an algorithm to find a feasible plan y_1, y_2, \dots, y_n which is closest to the input plan x_1, x_2, \dots, x_n . Also output the distance between x_1, x_2, \dots, x_n and y_1, y_2, \dots, y_n . Your goal is to implement an algorithm with as low time complexity as possible (to achieve this objective, you may use more space if required).

Your algorithm should run in $\mathcal{O}(n)$ time and $\mathcal{O}(n)$ space (other than the input) to get full marks for algorithm design part.

main()

1. Read n from the user.
2. Read x_1, x_2, \dots, x_n from the user. Use only malloc/calloc/new whenever you need to allocate memory dynamically
3. Output a feasible plan y_1, y_2, \dots, y_n which is closest to the input plan x_1, x_2, \dots, x_n

Submit a single .c or .cpp file. Your code should get compiled properly by gcc or g++ compiler.

Sample Output

Write n: 4

Write plan: 8 1 4 5

Closest feasible plan: 8 2 4 6

Distance: 2

Write n: 3

Write plan: 10 3 4

Closest feasible plan: 10 3 4

Distance: 0