Question 2

You are managing a garage with two mechanics, named Alice and Bob. Each mechanic can serve at most one customer at a time.

There will be n customers who come in during the day. The ith customer wants to be served for the entire period of time starting at time s_i and ending at time e_i. You may assume that the customers are indexed by their order of arrival, i.e. $s_1 < s_2 < \ldots < s_n$.

For each customer i, the business will:

- make a_i dollars if customer i is served by Alice;
- make b_i dollars if customer i is served by Bob;
- lose c_i dollars if customer i is not served.

Your task is to maximise the net earnings of the garage, which is calculated as the total amount made minus the total amount lost.

2.1 **[8 marks]** Consider the following greedy algorithm.

Process each customer i in order of arrival as follows.

- If both Alice and Bob are available at time s_i:

    - if $a_i \geq b_i$, assign customer i to Alice;

    - otherwise, assign the customer to Bob.

- If only one mechanic is available at time s_i, assign customer i to that mechanic.

- If neither mechanic is available at time s_i, do not serve customer i.

Design an instance of the problem which is not correctly solved by this algorithm. You must:

- specify a number of customers n,
- for each customer, provide values for s_i, e_i, a_i, b_i and c_i,
- apply the greedy algorithm to this instance and calculate the net earnings achieved, and
- show that a higher net earnings figure can be achieved.

> Answer:
>
> Suppose i is the index of any consumer in n who comes to the service, according to the topic, a_i dollars if customer i is served by Alice, b_i dollars if customer i is served by Bob, lose c_i dollars if customer i is not served, s_i is start time of the consumer i want to be served and e_i is the end time of the consumer i finish the work. Suppose A, B is the index of Alice and Bob is working or had worked for customer i, s_A the start time of Alice need to work. Suppose an array opt which contains the maximum money, $opt[i]$ means that the maximum money can earn when finish the customer i work.
>
> set $A = -1, B = -1$
> Now, perform the following step while $i \geq 1$ AND $i \leq n$:
>
> - if $A = -1$ OR $B = -1$:
>
>   - if $A = -1$ AND $B = -1$, $opt(i) = max\, a_i, b_i$

2.2 [**12 marks**] Design an algorithm which runs in $O(n^2)$ time and determines the maximum net earnings of the garage.

Your answer here.