

Exercice 1 :

Une route comporte n stations-service, numérotées dans l'ordre du parcours, de 0 à $n-1$. La première est à une distance $d[0]$ du départ, la deuxième est à une distance $d[1]$ de la première, la troisième à une distance $d[2]$ de la deuxième, etc. La fin de la route est à une distance $d[n]$ de la n -ième et dernière station-service. Un automobiliste prend le départ de la route avec une voiture dont le réservoir d'essence est plein. Sa voiture est capable de parcourir une distance r avec un plein.

Question 9.5.1

Donnez une condition nécessaire et suffisante pour que l'automobiliste puisse effectuer le parcours. On la supposera réalisée par la suite.

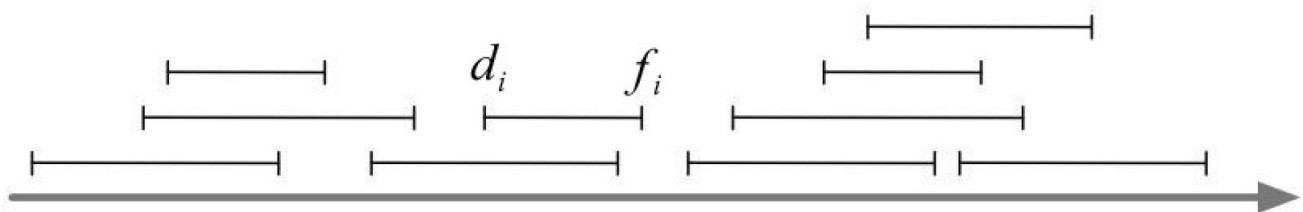
Question 9.5.2

Prenez 17 stations-service avec les distances $d = [23, 40, 12, 44, 21, 9, 67, 32, 51, 30, 11, 55, 24, 64, 32, 57, 12, 80]$ et $r = 100$.

L'automobiliste désire faire le plein le moins souvent possible. Écrivez en pseudo-code, puis programmez une fonction Python *rapide* qui détermine à quelles stations-service il doit s'arrêter.

Exercice 2 :

Dans un cinéma, chaque séance i est caractérisée par l'intervalle (d_i, f_i) , où d_i est l'heure de début et f_i l'heure de fin. On peut représenter ces séances sur un schéma d'intervalles :



Exemple de planning des séances de cinéma

Vous voulez assister au maximum de séances dans une journée.

Vous considérez trois critères pour classer les séances, de la plus petite valeur à la plus grande :

1. critère A : l'heure de début de la séance (d_i)
2. critère B : la durée de la séance ($f_i - d_i$)
3. critère C : l'heure de fin de la séance (f_i)

a. Décrivez en pseudo-code un algorithme glouton permettant de choisir les séances, après les avoir classées selon l'un des critères ci-dessus.

b. Pour chacun des trois critères de classement, exhibez un cas (s'il en existe un) où votre algorithme ne donnera pas un choix optimal. Donnez un exemple avec 5 séances sous forme d'un schéma d'intervalles comme celui de l'énoncé du problème.

c. Appliquez votre algorithme aux séances ci-dessous. Donner le résultat pour chacun des trois critères de classement.

Séance (i)	1	2	3	4	5	6	7	8	9	10
Début (d_i)	9h	9h15	10h	13h	15h	15h15	16h	17h30	18h	19h30
Fin (f_i)	11h	10h50	11h20	15h25	16h40	18h15	18h05	19h	20h10	22h
Durée ($f_i - d_i$)	120	95	80	145	100	180	125	90	130	150

d. Vous voulez absolument assister à la séance 9. Modifiez votre algorithme pour intégrer cette contrainte supplémentaire, puis répondez à nouveau à la question **c**.

Exercice 3 : https://pixees.fr/informatiquelycee/n_site/nsi_prem_knn.html