Sia $S = \{a_1, a_2, \ldots, a_n\}$ un insieme di n attività che competono per l'uso di una stessa risorsa R, ove ciascuna attività è caratterizzata da un tempo di inizio utilizzo s_i e un tempo di fine utilizzo f_i, tali che $0 \leq s_i < f_i < \infty$, per $i = 1, 2, \ldots, n$.

Si supponga inoltre che la risorsa R risulti indisponibile a causa di manutenzione durante gli intervalli temporali $[x_j, y_j]$, con $0 \leq x_j < y_j$ e $j = 1, 2, \ldots, m$.

(a) Si descriva un algoritmo che selezioni il sottoinsieme di S che contiene il maggior numero possibile di attività mutuamente compatibili e compatibili anche con gli intervalli temporali in cui R non è disponibile.

(b) Si illustri l'algoritmo trovato al punto (a) nel caso in cui s_i, f_i, x_j, y_j siano dati dalle seguenti tabelle:

i	1	2	3	4	5	6	7	8	9	10	11
s_i	7	6	16	3	13	18	4	1	12	15	12
f_i	10	8	21	5	16	20	6	2	14	17	14

j	1	2
x_j	7	18
y_j	8	19

Professor Midas drives an automobile from Newark to Reno along Interstate 80. His car's gas tank, when full, holds enough gas to travel n miles, and his map gives the distances between gas stations on his route. The professor wishes to make as few gas stops as possible along the way. Give an efficient method by which Professor Midas can determine at which gas stations he should stop, and prove that your strategy yields an optimal solution.

Describe an efficient algorithm that, given a set $\{x_1, x_2, ..., x_n\}$ of points on the real line, determines the smallest set of unit-length closed intervals that contains all of the given points. Argue that your algorithm is correct.

Suppose you are given two sets A and B, each containing n positive integers. You can choose to reorder each set however you like. After reordering, let a_i be the ith element of set A, and let b_i be the ith element of set B. You then receive a payoff of $\prod_{i=1}^{n} a_i^{b_i}$. Give an algorithm that will maximize your payoff. Prove that your algorithm maximizes the payoff, and state its running time.

ESERCIZIO 2

Si consideri il seguente testo $\mathcal{T}$ di 59 caratteri

$$\mathsf{SI␣SENTIVANO␣CONTINUI␣CORI␣DI␣POPOPOPOPOPOPOPO␣POPOPOPOPOPOPO}$$

ove il simbolo "␣" rappresenta il blank.

Dopo avere illustrato l'algoritmo di Huffman, si trovi un codice prefisso binario per l'alfabeto dei simboli occorrenti in $\mathcal{T}$ che ne minimizzi la dimensione e si calcoli il risparmio in percentuale realizzato rispetto ad una rappresentazione di $\mathcal{T}$ mediante una codifica a lunghezza fissa minima.

Nel contesto della metodologia *greedy*, si enunci il problema di ottimizzazione relativo alla *selezione di attività* e se ne discuta una soluzione efficiente, valutandone la complessità computazionale e illustrandola sull'insieme di attività $S = \{a_1, \ldots, a_{12}\}$, caratterizzate dai seguenti tempi iniziali e finali:

i	1	2	3	4	5	6	7	8	9	10	11	12
s_i	7	24	11	18	16	1	13	20	2	25	30	8
f_i	14	30	16	24	22	9	17	24	9	27	35	15

Sia T un testo di 250 caratteri nell'alfabeto $\{a_1, \ldots, a_{10}\}$, ove la frequenza di a_i è data dall'espressione $f_i = i^2 - 3i + 3$, per $i = 1, \ldots, 10$.

Dopo aver dato la definizione di *codice prefisso*, si stabilisca qual è il numero minimo di bit necessari per rappresentare il testo T utilizzando un codice prefisso ottimo.

a_1	1	7	7
a_2	1	7	7
a_3	3	6	18
a_4	7	5	35
a_5	13	4	52
a_6	21	3	63
a_7	31	3	93
a_8	43	3	129
a_9	57	2	114
a_{10}	73	2	146
			?!