

Esercizi - Programmazione C

Anno Accademico 2022/2023

Dott. Staccone Simone



TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Esercizi

Implementare in codice C i seguenti esercizi applicando le dovute accortezze:

- Modularità del codice
- Compilazione attraverso MakeFile
- Input possibilmente tramite argomenti del programma
- Tutto quello di cui abbiamo discusso ma che non mi viene in mente adesso!

Esercizi

- 1) Dato un array di interi noto a tempo di compilazione, trovare il massimo e il minimo.
- 2) Dato un array di interi noto a tempo di compilazione, trovare la media tra tutti i valori dell'array
- 3) Dato un array di interi noto a tempo di compilazione, trovare la dimensione della più lunga sotto sequenza di numeri crescente (Es. arr = [1,4,5,2,3,6,7,2,4,1,2] la risposta è 4 perché la sotto sequenza più lunga è [2,3,6,7])
- 4) Dato un array di interi noto a tempo di compilazione, trovare la più lunga sotto sequenza di numeri crescente (Es. arr = [1,4,5,2,3,6,7,2,4,1,2] la risposta è l'array [2,3,6,7])
- 5) Dato un array di interi trovare il massimo e il minimo (Nota bene che il programma andrebbe lanciato come ./[nome eseguibile] [arr[0]] [arr[1]] ... Non è noto il numero di parametri da passare a priori. Bisogna allocare il vettore in maniera dinamica)

Esercizio 1

Il carceriere e i dolcetti: Una carceriere ha un certo numero di prigionieri e una serie di dolcetti da distribuire loro. Il loro carceriere decide che il modo più equo per dividere i dolcetti è far sedere i prigionieri attorno a un tavolo circolare su sedie numerate in sequenza. Un numero di sedia verrà estratto da un cappello. A partire dal prigioniero su quella sedia, un dolcetto verrà distribuito a ciascun prigioniero in sequenza attorno al tavolo fino a quando non saranno stati distribuiti tutti. Il carceriere sta facendo uno scherzo, però. L'ultimo dolcetto è come tutte gli altri, ma ha un sapore orribile. Determina il numero della sedia occupata dal prigioniero che riceverà quel dolcetto.

Descrizione della funzione: La funzione deve restituire un numero intero che rappresenta il numero di sedia del prigioniero da avvertire.

n: il numero dei detenuti

m: il numero di dolci

s: il numero della sedia da cui iniziare a distribuire i dolci

Esempio: $n=4, m=6, s=2 \rightarrow$ La risposta è 3 (il prigioniero seduto nella terza sedia riceverà il dolcetto)

Test cases (n,m,s) \rightarrow soluzione

(5,2,1) \rightarrow 2; (5,2,2) \rightarrow 3; (7,19,2) \rightarrow 6; (3,7,3) \rightarrow 3

Esercizio 2

La regina degli scacchi: Data una scacchiera quadrata con una regina e una serie di ostacoli posizionati su di essa, determina quante caselle la regina può attaccare. Una regina è su una scacchiera $n \times n$. Le righe della scacchiera sono numerate da 1 a n , andando dal basso verso l'alto. Le sue colonne sono numerate da 1 a n , andando da sinistra a destra. Ogni quadrato è referenziato da una tupla (r,c) che descrive la riga r e la colonna c dove si trova il quadrato. La regina è in posizione (r_q, c_q) . Con una singola mossa, può attaccare qualsiasi casella in una qualsiasi delle otto direzioni (sinistra, destra, su, giù e le quattro diagonali). Nei diagrammi seguenti, i cerchi verdi indicano tutte le celle da cui la regina può attaccare

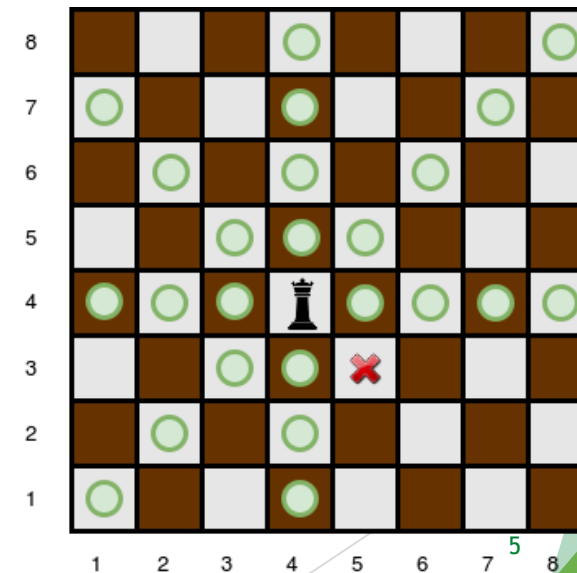
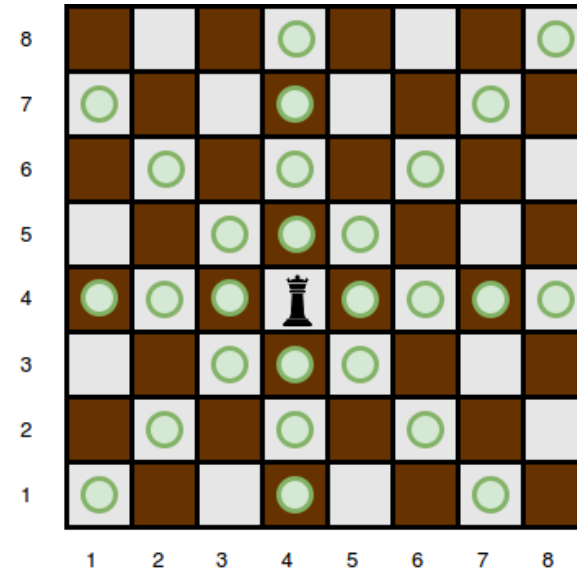
n : numero di righe e di colonne della scacchiera

k : numero di ostacoli sulla scacchiera

r_q : riga di posizione della regina

c_q : colonna di posizione della regina

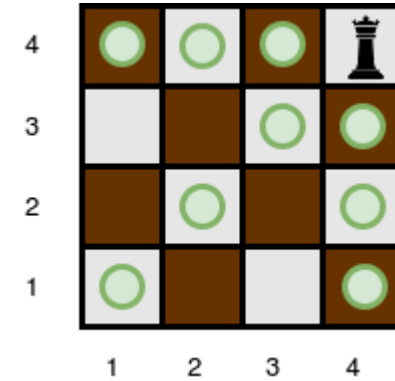
$obstacles[k][2]$: ogni element è un array di dimensione due contenente le coordinate degli ostacoli



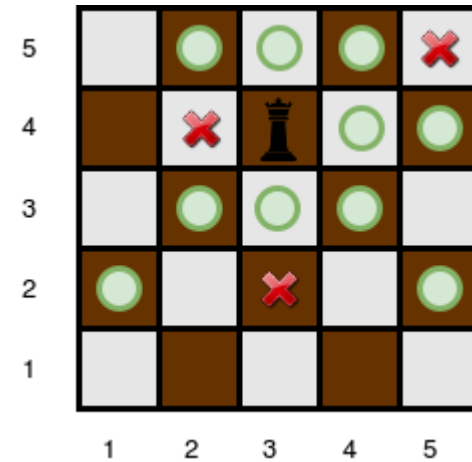
Esercizio 2 - 2

Esempi:

$n=4; k=0; c_q=4; r_q=4; obstacles=[]; \rightarrow 9$



$n=5; k=3; c_q=4; r_q=3; obstacles=[[5,5],[4,2],[2,3]]; \rightarrow 10$



$n=1; k=0; c_q=1; r_q=1; obstacles=[]; \rightarrow 0$