

Esercitazione di Fondamenti di Informatica del 03/12/2019

Esercizio 1.

Scrivere ed eseguire un programma C++ opportunamente suddiviso in funzioni che letta in input una sequenza di 10 numeri, calcoli e stampi su standard output:

- la media dei numeri letti (come divisione tra la somma dei numeri e 10)
- il numero massimo tra i numeri letti

Inoltre, siano ME la media e MA il massimo, per ogni numero X della sequenza, calcolare e stampare su standard output:

- la differenza ME - X
- la differenza MA - X

Esempio.

Data la sequenza 2 3 6 7 87 9 15 7 1 6, il programma dovrebbe stampare:

```
media: 14
max: 87
Differenze con media:
12
11
8
7
-73
5
-1
7
13
8
Differenze con max:
85
84
81
80
0
78
72
80
86
81
```

Per semplicità calcolare la media utilizzando la divisione intera. Nell'esempio infatti, la media è 14.3 ma viene stampato 14.

Suggerimento. Memorizzare la sequenza in un array di interi e realizzare (almeno) le seguenti funzioni:

per il calcolo della media:

int media(int[] array, int dim)

per il calcolo del massimo

int max(int[] array, int dim)

per il calcolo e la stampa delle differenze di ogni elemento dell'array con un intero N:

void diff(int[] array, int dim, int N)

Quest'ultima funzione potrà essere utilizzata sia per le differenze con la media sia con il massimo.

Esercitazione di Fondamenti di Informatica del 03/12/2019

Esercizio 2.

Scrivere ed eseguire un programma C++ opportunamente suddiviso in funzioni che letta in input una sequenza di 10 caratteri, stampi su standard output "SI" se la sequenza è palindroma, "NO" altrimenti.

Una sequenza di caratteri si dice palindroma se, letta al contrario, rimane invariata. Ad esempio, sono palindrome le sequenze: "abcbaabcba", "aeiouuoiea".

Esercizio 3.

Scrivere ed eseguire un programma C++ opportunamente suddiviso in funzioni che simuli il gioco Monty Hall. Il gioco Monty Hall è così chiamato dal nome del conduttore di un gioco televisivo in cui un concorrente può vincere un'auto indovinando la porta dietro la quale si trova l'auto. Il giocatore sceglie una porta, allora il conduttore ne apre un'altra dietro la quale non si trova l'auto. A questo punto del gioco, il giocatore può scegliere se cambiare la sua scelta con la terza porta. Il programma dovrà infine indicare se l'auto è stata vinta o meno.

Suggerimento. Si potrebbero rappresentare le porte con i numeri da 1 a 3. Per fare in modo che il gioco non sia sempre uguale, scegliere in modo casuale il numero della porta dietro la quale nascondere l'auto.

Più in dettaglio, il programma dovrebbe eseguire le seguenti operazioni:

1. Stampare la spiegazione del gioco.
2. Determinare la porta con dietro l'auto in base ad un numero random.
3. Chiedere al concorrente di scegliere una porta.
4. Aprire una porta differente da quella scelta, dietro la quale non si nasconde l'auto.
5. Proporre al concorrente il cambio.
6. Nel caso in cui il concorrente accetti, aggiornare la sua scelta.
7. Determinare il risultato finale del gioco.

Esercizio 4.

Scrivere in C++ un programma opportunamente suddiviso in funzioni che:

- legga da input una sequenza di 10 numeri interi;
- memorizzi in un array tutti i numeri della sequenza;
- scandisca l'array e stampi su standard output tutte le sotto-sequenze terminate da 2, nonché tutte le posizioni in cui compare un "2".

Per semplicità, si può assumere che l'ultimo intero della sequenza sia sempre "2".

Esempio.

Se la sequenza in input fosse la seguente:

3 6 8 8 2 4 6 2 6 2

si dovrebbero stampare su standard output le 4 sotto-sequenze separate dai "2":

{3 6 8 8}, {4 6}, {6}

ed infine le posizioni occupate dai "2", ovvero:

4, 7, 9