

Foglio esercizi 3

Esercizi sui puntatori.

Esercizio 1

Definire le variabili intere v_1 , v_2 ed i puntatori p_1 , p_2 , p_3 .

Svolgere le seguenti istruzioni:

1. Stampare i valori di v_1 e v_2 ;
2. Stampare gli indirizzi di v_1 e v_2 ;
3. Assegnare valori diversi a v_1 e v_2 e stamparli;
4. Stampare i valori delle variabili puntate da p_1 e p_2 ;
5. Stampare gli indirizzi di p_1 e p_2 ;
6. Assegnare a p_1 l'indirizzo di v_1 ed a p_2 l'indirizzo di v_2 ;
7. Stampare gli indirizzi salvati in p_1 e p_2 ;
8. Stampare i valori delle variabili puntate da p_1 e p_2 ;
9. Incrementare v_1 e stampare il suo valore;
10. Stampare l'indirizzo contenuto da p_1 ;
11. Stampare il valore della variabile puntata da p_1 ;
12. Assegnare a p_2 l'indirizzo di v_1 ;
13. Stampare il valore della variabile puntata da v_2 ;
14. Incrementare il valore della variabile puntata da p_1 , attraverso p_1 stesso;
15. Stampare il valore della variabile puntata da p_2 ;
16. Assegnare a p_3 l'indirizzo di p_2 (p_3 pointer a pointer);
17. Stampare il valore della variabile puntata da p_3 (puntata a sua volta da p_2), tramite p_3 ;
;
18. Decrementare il valore della variabile puntata da p_3 (puntata a sua volta da p_2),
tramite p_3 .

Commentare ogni istruzione spiegando cosa sta succedendo in termini di assegnazioni, indirizzi e locazioni di memoria.

Esercizio 2

Si scriva una funzione che, dati quattro numeri interi a, b, c, d , scambi i loro valori in modo che, una volta finita l'esecuzione della funzione, si abbia: $a \leq b \leq c \leq d$.

Gli interi devono essere passati alla funzione tramite puntatore e la funzione deve ritornare void. La funzione sarà definita come segue:

```
void ordine_decrescente(int* a, int* b, int* c, int* d)
```

Nota: puoi utilizzare la funzione `scambia` dell'Esempio 5.2 della dispensa, ed implementare una funzione `minimo` che calcoli il minimo tra due interi passati tramite puntatore:

```
int min(int* a, int* b)
```

Esercizio 3

Si scriva una funzione ricorsiva (possibilmente ricorsiva dal basso), che prenda come input un intero positivo n e il puntatore della variabile H , che conterrà il seguente risultato:

$$H(n) = \sum_{i=1}^n \frac{1}{i} = 1 + \frac{1}{2} + \dots + \frac{1}{n}$$

La funzione deve essere definita nel seguente modo:

```
void somma_armonica(int n, double* h)
```