

Cognome Nome Matricola

(Se si sta sostenendo un' integrazione, indicare da quanti crediti:)

Programmazione - Appello del 29/6/2022

Istruzioni

- Prima di iniziare, scrivete nome, cognome e matricola su *ogni foglio*. Se state sostenendo un'integrazione, indicatelo sopra.
- Dall'inizio della prova avete a disposizione 1h:30 (per quelli di voi che hanno diritto al 30% di tempo aggiuntivo 1:57h, 70 minuti per chi fa l'integrazione da 6-7 crediti) per consegnare il compito alla cattedra. Consegne in ritardo non verranno accettate. Se state facendo un'integrazione, dovete completare solamente gli esercizi con indicato (integrazione $\geq x$ crediti).
- Potete usare solamente penne, matita e gomma. Nient'altro può essere presente sul banco. Non potete usare fogli aggiuntivi, nemmeno per la brutta copia. Appoggiate zaini e giacche a lato della stanza. Scrivete in modo leggibile, parti non comprensibili non verranno corrette.
- Vi chiedo di non parlare e non alzare lo sguardo dal vostro foglio. Una volta finito il compito non utilizzate il cellulare. Visto che siete seduti vicini gli uni agli altri, sarò intransigente, segnandomi la minima trasgressione e riservandomi successivamente come agire.

1. Osservare i seguenti codici e rispondere alle domande sotto di essi, relative al valore di una variabile. In particolare specificare se il codice compila e quale valore assumono le variabili. Motivare la risposta.

```
1  #include <stdio.h>

3  int main(void) {
4      int a, b=2, *p, *q, **qq, **pp;
5      q = &a;
6      qq = &&b;
7  }
```

Qual è il valore di qq alla riga 7?

2 punti

Risposta: il codice non compila per un errore alla riga 6: l'operatore & può essere applicato solo ad una variabile puntatore, non ad un L-valore (come nel caso del & più a sinistra).

```
1  #include <stdio.h>

3  int main(void) {
4      int a, b=2, *p, *q, **qq, **pp;
5      q = &a;
6      pp = &p;
7      p=q;
8      **pp = 7;
9      printf("%d\n", a);
10     q=&b;
11     **pp = 3;
12     printf("%d\n", a);
13 }
```

Cosa stampa il seguente codice?

4 punti

Risposta: alla riga 9 il codice stampa 7; alla riga 12 il codice stampa 3.

Alla riga 9 il codice stampa 7 poiché alla riga 5 q prende l'L-valore di a ed alla riga 7 p prende anch'esso l'L-valore di a; infine pp alla riga 6 prende l'L-valore di p, per cui **pp va a modificare a, che prenderà il valore 7.

Alla riga 10 l'R-valore di q viene modificato ma questo non ha effetto su p, per cui **pp=3 andrà ancora a modificare la variabile a, per cui si stampa 3.

2. La funzione `rimuovi_triple()`, dato un array di interi `A`, elimina tutti gli elementi di `A` uguali al precedente e al successivo (il primo e l'ultimo elemento non verranno mai eliminati).

Il tipo restituito dalla funzione è `void`. I parametri formali della funzione sono l'array `A` contenente gli interi e la sua dimensione `dim` (un intero).

Dopo l'invocazione della funzione, i parametri attuali corrispondenti ad `A` e `dim` (uso gli stessi nomi per semplicità) devono rispettare le seguenti condizioni:

- `dim` equivale al numero di elementi superstiti (non cancellati) dell'array `A`.
- `A` deve contenere nelle prime `dim` posizioni i valori superstiti.

La funzione non deve operare su altri array oltre ad `A`. Esempi di input/output:

- `A=1,1,1,2,6,5` diventa `A=1,1,2,6,5`
- `A=2,1,1,1,1,2,6,5` diventa `A=2,1,1,2,6,5`

L'esercizio consiste nel

1. scrivere la PRE della funzione **1 punto**
2. Implementare la funzione `rimuovi_triple()` secondo la consegna. **8 punti** (integrazione ≥ 6 crediti)

```
1 void rimuovi_triple(int *A, int *dim) {
2     // PRE: *dim equivale alla dimensione di A
3     int i=1,j;
4     while(i<*dim-1) {
5         if ( (A[i]==A[i-1]) && (A[i]==A[i+1]) ) {
6             for(j=i;j<*dim-1;j++) /* "eliminiamo" l'elemento in
                                   posizione i spostando tutti gli elementi alla sua
                                   destra di una posizione a sinistra */
7                 A[j] = A[j+1];
8             *dim -= 1;
9         } else
10             i+=1;
11     }
13 }
```