

Fondamenti di Informatica - A.A. 2025-2026

Scuola di Ingegneria Industriale e dell'Informazione
Prof.ssa Cristiana Bolchini
Appello del 28/01/2026

RESTITUIRE COMPILATO



POLITECNICO
MILANO 1863

Cognome

Nome

Cod. Persona

Quesito:	1	2	3	4	5	6	Totale
Valutazione massima (in /30):	1	4	5	6	7	7	30
Valutazione quesito in decimi (/10):							

Istruzioni:

- gli esercizi devono essere risolti utilizzando il C ANSI 89, in linea con quanto fatto durante il corso;
- non è possibile consultare libri, appunti, la calcolatrice o qualsiasi dispositivo elettronico, tantomeno comunicare;
- si può scrivere con qualsiasi colore, anche a matita, pur di scrivere con un tratto leggibile (calcare bene).
- è consentito ritirarsi; nel caso si consegna una prova valutata meno di 6/30 (insufficienza grave) non è consentito iscriversi all'appello successivo
- tempo a disposizione: 1h 40m

Stile del codice C:

- non è necessario inserire direttive #include;
- i commenti non sono necessari, ma potrebbero essere utili nel caso di errore;
- è possibile utilizzare sottoprogrammi di libreria.
- è possibile organizzare il proprio codice in sottoprogrammi non esplicitamente

**SCRIVERE SOLO
ALL'INTERNO DELLE CORNICI**

Queste informazioni non vengono in alcun modo considerate ai fini della valutazione

Sapevo già programmare:

No, non è vero in C in C++/C# in Python in Java in PHP/Javascript in VB* in altro linguaggio

Ho frequentato (lezioni ed esercitazioni):

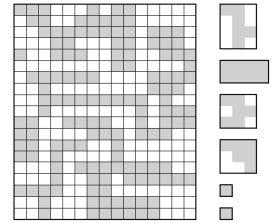
occasionalmente fino alle prove in itinere (metà corso) fino alla fine ma con molte assenze sempre (occasionalmente assenze)

Sottoprogrammi per la gestione di liste concatenate semplici: qualora utili, si considerino già disponibili (e quindi non da sviluppare). Qua ci si riferisce a un campo intero, ma nel caso la lista debba gestire caratteri (o dati di natura diversa), ipotizzate di avere a disposizione il sottoprogramma corrispondente analogo. **Definite il tipo di dato listtype così come vi è utile, e mettete tale definizione all'inizio dell'esercizio.**

```
/* inserisce in testa alla lista elemento con valore specificato */
listtype * push(listtype *, int);
/* inserisce in coda alla lista elemento con valore specificato */
listtype * append(listtype *, int);
/* inserisce ordinatamente in lista elemento con valore specificato, in senso crescente */
listtype * increasing(listtype *, int);
/* inserisce ordinatamente in lista elemento con valore specificato, in senso decrescente */
listtype * decreasing(listtype *, int);
/* elimina dalla lista il primo elemento */
listtype * pop(listtype *);
/* elimina dalla lista tutti gli elementi con il valore indicato */
listtype * delete(listtype *, int);
/* restituisce il riferimento all'elemento che ha il valore indicato, se esiste, NULL altrimenti */
listtype * find(listtype *, int);
/* restituisce il numero di elementi nella lista */
int length(listtype *);
/* elimina la lista */
listtype * emptylist(listtype *);
```


Quesito 5 [7 pti]

Si consideri un gioco da tavolo costituito da un tabellone bidimensionale sul quale devono essere collocati pezzi di varie forme esclusivamente nelle posizioni libere (in modo analogo alla costruzione di un puzzle). In una realizzazione informatica del gioco, si rappresenta il tabellone mediante un array bidimensionale di caratteri: in ciascuna cella il carattere 'X' indica una posizione occupata, mentre il carattere ' ' indica una posizione libera. Analogamente, un pezzo è rappresentato mediante un array bidimensionale, in cui 'X' indica una parte del pezzo e ' ' una cella vuota.



Scrivere un sottoprogramma che riceve in ingresso due array bidimensionali di caratteri `tab` e `pezzo`, e qualunque ulteriore parametro ritenuto strettamente necessario. Il sottoprogramma verifica se il pezzo può essere posizionato sul tabellone `tab` andando ad occupare solo celle libere: il sottoprogramma restituisce 1 in caso affermativo, 0 in caso negativo. L'array `pezzo` è dimensionato per contenere esclusivamente il pezzo relativo alla mossa corrente, in modo tale che non ci siano righe o colonne completamente vuote e la dimensione del pezzo è tale per cui ci sta senz'altro sul tabellone, a meno di problemi di parti sovrapposte (ovvero non è necessario controllare la compatibilità delle dimensioni di tabellone e pezzo).

Sono preseti le seguenti direttive per le dimensioni massime degli array bidimensionali

```
#define NCOL_MAP /* numero massimo di colonne dell'array del tabellone */
#define NROW_MAP /* numero massimo di righe dell'array del tabellone */
#define NCOL_PIECE /* numero massimo di colonne dell'array del pezzo */
#define NROW_PIECE /* numero massimo di righe dell'array del pezzo */
```

Esempio:

- Con i seguenti array in ingresso (ho sostituito il carattere nel secondo array per motivi di leggibilità) il sottoprogramma restituisce 1.

```

      XX X
      XXX XX
tab:  XXX X
      X X X
      XXXXX
           00
pezzo:  0
           00
           00
posizione:  XX00X
            XXX0XX
            XXX00X
            X X X
            XXXXXX
```

- Con i seguenti array in ingresso il sottoprogramma restituisce 0.

```

      XX X
      XXX XX
tab:  XXX X
      X X X
      XXXXX
           00
pezzo:  0
           00
           0
```

Quesito 6 [7 pti]

(6 pti) Scrivere un programma che riceve da riga di comando il nome di un file di testo contenente una sequenza di valori interi. Il programma visualizza il valore che compare più di frequente nel file.

Esempio: Contenuto del file `dati.txt`

```
12 3 7 1 8 3 9 0 -4 1 3 5 1 6 1 9 0 -2 3 1 7 3 19 0 2 0 3 7
```

Esempio di esecuzione:

```
./exe4 dati.txt
```

```
3
```

(1 pto) Motivare la scelta della struttura dati adottata.

