

# Fondamenti di Informatica - A.A. 2022-2023

Scuola di Ingegneria Industriale e dell'Informazione  
Prof.ssa Cristiana Bolchini  
Appello del 01/02/2023



POLITECNICO  
MILANO 1863

Cognome	Nome	Matricola o Cod. Persona
---------	------	--------------------------

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

INIZIARE LA SOLUZIONE DI OGNI  
ESERCIZIO SU UNA PAGINA NUOVA

RESTITUIRE COMPILATO ANCHE  
NEL CASO IN CUI CI SI RITIRA

### 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, né comunicare;
- si può scrivere con qualsiasi colore, anche a matita, ad eccezione del rosso.
- 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.

### Sapevo già programmare:

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

### Quesito 1 [3 pts]

Data la sequenza 11001, dire a quale valore intero espresso in base dieci corrisponde, nel caso esso sia rappresentato:

- valore naturale base 10: \_\_\_\_\_
- valore naturale base 16: \_\_\_\_\_
- valore naturale base 2: \_\_\_\_\_
- valore intero base 2, notazione modulo e segno: \_\_\_\_\_
- valore intero base 2 notazione complemento a 2: \_\_\_\_\_

Svolgere i calcoli sul foglio di protocollo e riportare qua i risultati, utilizzando solo parole. Per esempio volendo riportare il risultato +24 scrivere positivo ventiquattro, volendo riportare -11, scrivere negativo undici.

### Quesito 2 [1 pts]

Effettuare direttamente qua l'operazione tra i due operandi, entrambi rappresentati in complemento a due, spiegando se si verifica overflow o meno.

11110101000 +  
10110101111 =  
\_\_\_\_\_

Si verifica overflow perchè \_\_\_\_\_

Non si verifica overflow perchè \_\_\_\_\_

### Quesito 3 [6 pts]

Scrivere un sottoprogramma che riceve in ingresso due valori interi senz'altro strettamente positivi; il sottoprogramma calcola e restituisce un nuovo intero ottenuto alternando le cifre dei due numeri a partire dalle meno significative. Nel caso un valore in ingresso abbia più cifre dell'altro, le rimanenti cifre vengono appese a sinistra del valore risultante. Non ci si preoccupi di eventuali problemi di overflow. Si parta con la cifra del primo dei due valori ricevuti in ingresso. Esempio: dati i due valori 121 e 35365, il sottoprogramma restituisce 35316251.

Non si facciano assunzioni sulle dimensioni dei dati in ingresso.

#### Quesito 4 [7 pts]

- (4 pts) Scrivere un sottoprogramma che riceve in ingresso una stringa costituita da un numero di caratteri senz'altro pari, calcola e restituisce al chiamante il carattere che compare solamente o nella prima metà della stringa o nella seconda metà della stringa. Tale carattere esiste senz'altro ed è unico. Per esempio, se il sottoprogramma riceve in ingresso `jsMqZzHRSfrFNqRNjqzRMjGDGLrsFMffZqzSjrRLrGFZHjNsSL` il sottoprogramma restituisce il carattere `D`.
- (3 pts) Scrivere un programma che acquisisce come argomenti da riga di comando una sequenza di stringhe (non se ne conosce la quantità) e per ciascuna di esse visualizza il carattere che compare solo nella prima o nella seconda metà della stringa.

#### Quesito 5 [7 pts]

Scrivere un sottoprogramma `codifica` che riceve in ingresso due array di interi, `valori` e `codice` e qualsiasi altro parametro ritenuto strettamente necessario. Ogni intero contenuto nell'array `codice` indica di quanti passi far scorrere a destra (se positivo) o sinistra (se negativo) il contenuto dell'array `valori`. Il sottoprogramma calcola e restituisce la somma dei valori che risultano essere all'inizio dell'array (indice 0) a valle di ogni rotazione. Per esempio, se il sottoprogramma riceve in ingresso array così popolati:

```
valori: 3 1 -4 2 0 5
codice: -2 3 1 8
```

il sottoprogramma effettua la seguente manipolazione:

```
-4 2 0 5 3 1 (codice -2, scorrimento a sinistra di 2 passi): valore -4
5 3 1 -4 2 0 (codice 3, scorrimento a destra di 3 passi): valore 5
0 5 3 1 -4 2 (codice 1, scorrimento a destra di 1 passi): valore 0
-4 2 0 5 3 1 (codice 8, scorrimento a destra di 8 passi, equivalente a 2 passi): valore -4
```

e restituisce il valore `-3 (-4 + 5 + 0 -4)`

#### Quesito 6 [6 pts]

Scrivere un sottoprogramma che riceve in ingresso una lista di interi e la modifica sostituendo a ciascuna coppia di elementi consecutivi la loro somma. Il sottoprogramma restituisce la lista modificata. Notare che se il numero di elementi della lista è dispari, l'ultimo elemento della lista rimane inalterato.

Esempio:

```
ingresso: 21 → -3 → 12 → 11 → 3 → -9 → 5 → |
uscita: 18 → 23 → -6 → 5 → |
```

Si considerino già disponibili (e quindi non da sviluppare) i sottoprogrammi seguenti, validi per qualsiasi tipo di lista che gestisca un campo intero:

```
/* 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 *);
```