

Fondamenti di Informatica - A.A. 2020-2021

Scuola di Ingegneria Industriale e dell'Informazione
Prof.ssa Cristiana Bolchini
Appello del 19/07/2021



POLITECNICO
MILANO 1863

Cognome

Nome

Matricola o Cod. Persona

Quesito:	1	2	3	4	Totale
Valutazione massima:	5	9	7	9	30
Valutazione in decimi (/10):					

Istruzioni:

- 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 30m

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.

Quesito 1 [5 pts]

(1 pts) Dato il valore $A = 1111100001$ lo si consideri rappresentato in base 2 notazione in complemento a 2 e lo si rappresenti sul numero minimo di bit strettamente necessario.

(1 pts) Dato il valore $A = 1111100001$ lo si consideri rappresentato in base 2 notazione modulo e segno e lo si rappresenti sul numero minimo di bit strettamente necessario.

(1 pts) Dato il valore $A = 1000000001$ lo si consideri rappresentato in base 2 notazione in complemento a 2 e si calcoli e scriva il suo opposto $-A$, utilizzando il numero minimo di bit strettamente necessari.

(1 pts) Rappresentazione dei numeri reali in base 2, notazione IEEE 754: dato il valore $10011110111111111111111011000000$, il suo valore in modulo è un valore

- ☐ incluso nell'intervallo (0,1]
☐ incluso nell'intervallo (1,2]
☐ incluso nell'intervallo (2,+inf)

Perché:

(1 pts) Rappresentazione dei numeri reali in base 2, notazione IEEE 754: dato il valore $110000000000000000000000000000$ convertirlo in base 10, notazione modulo e segno.

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 2 [9 pti]

Un elemento di una sequenza si dice **basso** se il suo valore è **strettamente minore del valore dell'elemento che lo precede e di quello che lo segue**; il primo e l'ultimo elemento della sequenza, non avendo entrambi gli elementi precedente e successivo non sono mai definiti **bassi**. **Si utilizzino le strutture dati più opportune per un uso non sovrabbondante della memoria.**

- (5 pti) Scrivere un sottoprogramma **cercabassi** che ricevuto in ingresso un **array di valori reali** (e qualsiasi altro parametro ritenuto strettamente necessario) **restituisce al chiamante gli elementi bassi trovati nell'array e trasmette inoltre il numero di tali elementi.**

Per esempio, se i valori ricevuti in ingresso sono

3.4 6.2 2.4 2.0 1.8 7.2 -3.1 -3.1 9.4 2.6 2.2 8.4 -3.1 5.4 2.1 7.4 -4.0 -11.4 -8.9

il sottoprogramma rileva i seguenti 5 elementi **bassi**:

1.8 2.2 -3.1 2.1 -11.4

- (4 pti) Scrivere un programma che chiede all'utente quanti **dati reali deve processare e intende fornire in ingresso**, un valore strettamente positivo, e fino a quando l'utente non fornisce una risposta valida la **richiesta viene re-iterata**, per poi procedere all'acquisizione. Il programma calcola gli elementi **bassi** avvalendosi del sottoprogramma **cercabassi**, quindi chiama il sottoprogramma **analizzabassi** (il cui prototipo è riportato di seguito) che si occupa di effettuare un'analisi dei dati e la loro visualizzazione grafica con librerie apposite. Terminata l'analisi il programma principale si conclude.

```
void analizzabassi(float [], int);
```

Quesito 3 [7 pti]

Dato un array bidimensionale di valori interi positivi, ogni punto rappresenta la quota di un vulcano in una area. Quando un vulcano erutta il cratere si apre ed esce la lava per cui la sua quota cresce di 2, e la lava che fuoriesce fa crescere di 1 anche la quota delle otto posizioni attorno, se hanno una quota inferiore a quella del vulcano *prima* dell'eruzione (la lava scende ...). I due array di seguito riportano la situazione prima e dopo l'eruzione dell'elemento in grassetto.

3 0 6 1 5 1 5 6 8 1	3 0 6 2 6 1 5 6 8 1
0 0 5 6 0 5 4 0 5 6	0 0 6 8 1 5 4 0 5 6
1 0 8 1 2 6 4 6 5 6	1 0 8 2 3 6 4 6 5 6
5 1 8 6 0 0 0 0 0 0	5 1 8 6 0 0 0 0 0 0
3 1 3 0 4 1 2 1 5 6	3 1 3 0 4 1 2 1 5 6
5 1 5 1 3 1 1 0 7 1	5 1 5 1 3 1 1 0 7 1
5 1 5 1 0 0 5 6 2 3	5 1 5 1 0 0 5 6 2 3
3 1 8 1 0 0 0 0 0 0	3 1 8 1 0 0 0 0 0 0
prima	dopo

- (5 pti) Scrivere un sottoprogramma che ricevuto in ingresso un array bidimensionale di interi, le coordinate del vulcano che erutta, e qualsiasi altro parametro ritenuto strettamente necessario, modifica l'array iniziale aggiornandolo con la configurazione delle quote corrispondenti a dopo l'eruzione.

Il numero di colonne dell'array bidimensionale dichiarato dal chiamante è specificato mediante una direttiva **define** del simbolo **NC**.

- (2 pti) Scrivere un sottoprogramma **visualizzamappa** che ricevuto in ingresso un array bidimensionale di interi e qualsiasi altro parametro ritenuto strettamente necessario visualizza la mappa delle quote (come quella prima riportata).

Quesito 4 [9 pti]

- (5 pti) Scrivere un sottoprogramma **analizza** che riceve come parametro una stringa contenente soltanto lettere minuscole dell'alfabeto (è senz'altro così). Il sottoprogramma identifica e **trasmette** al chiamante la prima lettera nell'ordinamento alfabetico che compare nella stringa e quante volte vi compare.

Per esempio, se il sottoprogramma riceve in ingresso la stringa **ciocco** il sottoprogramma identifica la lettera **c** che compare 3 volte.

- (4 pti) Scrivere un programma che per ogni vocabolo presente nel file di testo **parole.txt** visualizza il vocabolo e la lettera dell'alfabeto identificata dal sottoprogramma **analizza**, e il numero di volte che compare nel vocabolo. Il programma visualizza infine il numero massimo di occorrenze rilevato. I vocaboli contenuti nel file sono senz'altro di al più 30 caratteri.

Per esempio, se il file contiene il seguente testo

questo esercizio tratta di file di testo e stringhe
il programma visualizza:

e 1
c 1
a 2
d 1
e 1
e 1
e 1
e 1
2