

Fondamenti di Informatica - A.A. 2020-2021

Scuola di Ingegneria Industriale e dell'Informazione
Prof.ssa Cristiana Bolchini
Appello del 25/06/2021



POLITECNICO
MILANO 1863

Cognome

Nome

Matricola o Cod. Persona

Quesito:	1	2	3	4	Totale
Valutazione massima:	5	8	8	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 pti]

Domande vero/falso sulla rappresentazione dell'informazione

(1 pto) Il **principale** vantaggio della notazione in complemento alla base è (una sola risposta):

- ☐ l'immediatezza delle operazioni aritmetiche
- ☐ un più ampio insieme di valori rappresentabili a parità di numero di bit utilizzati
- ☐ la non ridondanza della rappresentazione del valore 0
- ☐ un più semplice metodo di calcolo dell'overflow nell'aritmetica

(1 pto) Dati i valori $A = 10010101010_{2C2}$ e $B = 11110101010_{2C2}$

- ☐ $A > B$
- ☐ $A < B$

(1 pto) Il valore $A = 10011101011_{2C2}$ è rappresentato utilizzando il numero minimo di bit strettamente necessari

- ☐ Vero
- ☐ Falso

(1 pto) Rappresentazione dei numeri reali in base 2, notazione IEEE 754: il valore del campo esponente è rappresentato in complemento a 2, per una efficace manipolazione degli esponenti nelle operazioni di prodotto

- ☐ Vero
- ☐ Falso

(1 pto) Rappresentazione dei numeri reali in base 2, notazione IEEE 754: nessun valore viene mai esattamente rappresentato, si commette sempre un errore di approssimazione: nella notazione in singola precisione l'errore commesso è più piccolo di quello in doppia precisione.

- ☐ Vero
- ☐ Falso

INIZIARE LA SOLUZIONE DI OGNI
ESERCIZIO SU UNA PAGINA NUOVA

RESTITUIRE COMPILATO ANCHE
NEL CASO IN CUI CI SI RITIRA

Quesito 2 [8 pt]

Dato un array bidimensionale di valori interi piena solo di 0 e 1 (è senz'altro così), definisco **segmento orizzontale** una successione di 1 consecutivi nella stessa riga. La **lunghezza del segmento** è il numero di 1 contenuti, l'inizio del segmento sono le coordinate **i e j** dell'elemento con j minimo della sequenza di 1. Vogliamo calcolare la **lunghezza e l'inizio del segmento di lunghezza massima** e se ce ne fosse più di uno di pari lunghezza massima, si prende il **primo**.

Scrivere un sottoprogramma che ricevuto in ingresso un array bidimensionale di interi, e qualsiasi altro parametro ritenuto strettamente necessario, **calcola e trasmette al chiamante** la posizione di inizio del segmento e la sua lunghezza.

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

Per esempio, dato l'array di seguito riportato, il sottoprogramma individuerrebbe il segmento evidenziato, trasmettendo al chiamante i valori 1, 3 e 5, corrispondenti alle coordinate 1, 3 primo elemento del segmento, e 5 la sua lunghezza.

```
1 0 1 1 1 0 1 1
0 0 0 1 1 1 1 1
1 1 1 0 0 0 1 1
1 1 1 1 1 0 0 0
1 1 1 0 0 1 1 1
1 0 0 0 0 1 1 1
1 0 1 1 0 1 1 0
```

Quesito 3 [8 pt]

(5 pt) Scrivere un sottoprogramma **listapicchi** che ricevuta in ingresso una lista ne restituisce una nuova costituita da tutti e soli gli elementi della prima lista seguiti solo da elementi con un valore strettamente inferiore a quello dell'elemento stesso. L'ultimo elemento della lista viene messo nella nuova lista creata.

Per esempio, se la lista in ingresso è

1 -> 5 -> 13 -> 11 -> 11 -> 4 -> -3 -> 1 -> -5

la lista restituita conterrà:

13 -> 11 -> 4 -> 1 -> -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 *);
```

(1 pt) Definire un tipo di dato opportuno **listtype** per la gestione di liste concatenate semplici per memorizzare un valore intero.

(2 pt) Scrivere una variante del sottoprogramma **listapicchi** che non inserisce nella lista dei picchi l'ultimo elemento della lista ricevuta in ingresso.

Quesito 4 [9 pt]

Si vuole realizzare le funzionalità base di un correttore ortografico che, mediante l'ausilio di un dizionario di parole salvato in un file di testo, identifica eventuali errori di battitura che hanno causato l'aggiunta di lettere errate all'interno di una parola o il loro mancato inserimento. Esempio: l'utente al posto di **ciao** digita **cxiauo** o **cia**.

(4 pt) Scrivere un sottoprogramma **istypo** che riceve due stringhe in ingresso, digitata (**dig**) e un vocabolo corretto (**voc**), e le confronta, restituendo 1 se tutti i caratteri di **voc** compaiono in **dig** nella stessa sequenza, eventualmente con ulteriori caratteri, 0 altrimenti. A titolo di esempio si considerino le seguenti casistiche:

- digitata = **cciao**, vocabolo = **ciao**: restituisce 1
- digitata = **caio**, vocabolo = **ciao**: restituisce 0
- digitata = **acciao**, vocabolo = **ciao**: restituisce 1
- digitata = **cia**, vocabolo = **ciao**: restituisce 0

(5 pti) Scrivere un programma che acquisisce da riga di comando due stringhe. La prima stringa contiene un vocabolo scritto in modo errato, la seconda il nome del file contenente il dizionario da consultare (i vocaboli contenuti sono di al più 35 caratteri). Il programma si avvale del sottoprogramma `istypo` per visualizzare la lista dei vocaboli che forse l'utente avrebbe voluto scrivere invece di quello contenente l'errore. Il programma visualizza anche il numero di vocaboli trovati. Per esempio, se si esegue il programma con il vocabolo `cciao` e `dizionario.txt` ciò che viene visualizzato è quanto segue:

```
./suggerisci acciaccio dizionario.txt
a
acca
accia
acciacco
acciai
acciaio
accio
ai
aia
aio
cacci
caccio
cacio
caco
caio
ci
ciao
ciccio
cio
i
io
o
22 alternative
```

Sapevo già programmare:

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