

Fondamenti di Informatica - A.A. 2017-2018

Scuola di Ingegneria Industriale e dell'Informazione
Prof.ssa Cristiana Bolchini
Appello del 26/01/2018



POLITECNICO
MILANO 1863

Cognome	Nome	Matricola	Voto: ... /30
---------	------	-----------	---------------

Quesito:	1	2	3	4	5	6	Tot.
Max:	5	5	6	6	6	2	30
Punti:							

INIZIARE LA SOLUZIONE DI OGNI
ESERCIZIO SU UNA PAGINA NUOVA

RESTITUIRE COMPILATO ANCHE
NEL CASO IN CUI CI SI RITIRA

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 45m

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 punti)

Punteggio ottenuto: .../5

Dati i due numeri $A = -58_{10MS}$ e $B = +58_{16MS}$ effettuare la conversione in base 2, notazione complemento a 2 (2C2), sul numero minimo di bit necessari a rappresentare entrambi gli operandi. Si effettuino quindi le operazioni $A+B$ e $A-B$ indicando esplicitamente se si verifica overflow o meno, e motivando la risposta. Mostrare i passaggi fatti.

Quesito 2 (5 punti)

Punteggio ottenuto: .../5

Scrivere un programma che riceve in ingresso una sequenza di valori interi positivi a priori di lunghezza ignota e che si ritiene terminata quando l'utente inserisce un valore negativo o nullo. Il programma calcola per ogni valore della sequenza il numero di divisori propri (sono esclusi 1 e il numero stesso) e salva sul file di testo ASCII `ris.txt` le coppie *valore numero-di-divisori-propri*, una per riga. Il programma al termine visualizza il numero di coppie salvate su file. Per esempio, se l'utente inserisce i valori 12 8 13 100 32 11 4 35 27 8 -1 il programma crea un file il cui contenuto è riportato di seguito, e visualizza il valore 10.

```
12 4
8 2
13 0
100 7
32 4
11 0
4 1
35 2
27 2
8 2
```

Quesito 3 (6 punti)

Punteggio ottenuto: .../6

Scrivere un sottoprogramma che riceve in ingresso un array bidimensionale quadrato di dimensione 5 (e qualsiasi altro parametro ritenuto strettamente necessario) e verifica che in ogni riga ed in ogni colonna ci siano sempre tutti e soli i valori da 1 a 5, estremi inclusi, e restituisce 1 se la proprietà sussiste, 0 altrimenti.

Quesito 4 (6 punti)

Punteggio ottenuto: .../6

Scrivere un sottoprogramma `checkISBN10` che riceve in ingresso una stringa che rappresenta un codice ISBN-10 (International Standard Book Number) di 10 cifre numeriche separate da `-`, utilizzato per identificare univocamente un volume prima del 2007 (dal 2007 in poi il numero è di 13 cifre). Il sottoprogramma restituisce 1 se il codice ISBN è valido, 0 altrimenti. Un codice ISBN-10 è valido se la somma delle somme è un multiplo di 11. La *somma delle somme* si calcola addizionando ogni cifra del codice alla somma delle precedenti cifre.

Esempi:

Ingresso:	0-306-40615-2	0-07-881809-4
Cifre:	0 3 0 6 4 0 6 1 5 2	0 0 7 8 8 1 8 0 9 4
Somma delle cifre:	0 3 3 9 13 13 19 20 25 27	0 0 7 15 23 24 32 32 41 45
Somma delle somme:	132	219
Uscita:	1	0

(1 punto dei 6)

Scrivere il programma che acquisisce **da riga di comando** la stringa dell'ISBN-10 e - utilizzando il sottoprogramma `checkISBN10` - visualizza 1 se il codice è corretto, 0 altrimenti.

Quesito 5 (6 punti)

Punteggio ottenuto: .../6

Scrivere un sottoprogramma che ricevuta in ingresso una lista `head` per la gestione di valori interi, un intero `val` ed un intero `limit`, elimina dalla lista tutti gli elementi che hanno `val` come valore intero se e solo se il numero di volte che questo compare nella lista è strettamente inferiore a `limit`. Definire il tipo di dato per la realizzazione di liste per la gestione di valori interi `list_t`. Ci si avvalga del sottoprogramma `delete` che ricevuta in ingresso una lista ed un valore intero, elimina dalla lista tutti gli elementi che hanno nel campo informazione il valore ricevuto come secondo parametro. **Il sottoprogramma deve essere sviluppato (2 punti).**

Quesito 6 (2 punti)

Punteggio ottenuto: .../2

Si consideri il seguente stralcio di codice:

```
#define N 100
...
int val;
int v[N];
int * dv;
list_t * head = NULL;
...
dv = (int *)malloc(N * sizeof(int));
...
/* crea una lista costituita da N elementi */
head = fill_list(N);
...
```

Indicare quanta memoria richiede la gestione dei dati mediante le tre diverse variabili `v`, `dv` e `head` considerando che in tutti i casi si gestiscono `N` interi, formulando il tutto mediante l'uso del costrutto `sizeof`;

<i>variabili</i>	<i>memoria complessiva utilizzata per la gestione di N elementi</i>	<i>commento</i>
<code>val</code>	<code>sizeof(int)</code>	viene proposto a titolo di esempio di come formulare la risposta
<code>v</code>	-	-
<code>dv</code>	-	-
<code>head</code>	-	-