

Fondamenti di Informatica - A.A. 2011-2012

Ingegneria Informatica
Facoltà di Ingegneria dell'Informazione
Prof.ssa Cristiana Bolchini
Appello del 01/03/2012

RESTITUIRE
COMPILATO



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

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

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: 2h 00m

Stile del codice C:

- non è necessario inserire direttive `#include`;
- i commenti non sono necessari, ma potrebbero essere utili nel caso di errore;
- è consentito l'utilizzo di funzioni di libreria.

Quesito 1 (5 punti)

Punteggio ottenuto: ... /5

Dati i due numeri $A = -A3_{16}$ e $B = +99_{10}$ effettuare la conversione in base 2, notazione complemento a 2, sul numero minimo di bit necessari a rappresentare gli operandi. Si effettuino poi, in tale rappresentazione, 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 chiede all'utente di inserire il nome di un file (al più 30 caratteri, incluso percorso ed estensione) il cui contenuto consiste di righe di al più 80 caratteri. Il programma visualizza sullo standard output tutte le righe che iniziano con un carattere maiuscolo. Gestire opportunamente i casi d'errore nell'accesso al file.

Quesito 3 (5 punti)

Punteggio ottenuto: ... /5

Scrivere un sottoprogramma che riceve in ingresso una stringa che rappresenta un codice ISBN (International Standard Book Number) utilizzato per identificare univocamente un volume. Il sottoprogramma restituisce 1 se il codice ISBN è valido, 0 altrimenti. Un codice ISBN 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.

Esempio:

Ingresso: 0-07-881809-5
Somma delle cifre: 0 0 7 15 23 24 32 32 41 46
Somma delle somme: 0 0 7 22 45 69 101 133 174 220
Uscita: 1



Quesito 4 (5 punti)

Punteggio ottenuto: ... /5

Scrivere un sottoprogramma che riceve in ingresso una matrice di valori interi e qualsiasi altro parametro ritenuto strettamente necessario, calcola e restituisce l'indice della colonna in cui la massima differenza ottenuta tra due suoi elementi, sia minima.

Quesito 5 (5 punti)

Punteggio ottenuto: ... /5

Scrivere un sottoprogramma che riceve in ingresso una lista di interi e una seconda lista dove ogni nodo rappresenta un valore intero e la sua frequenza (definire i due tipi opportuni). Per ogni nodo nella seconda lista, il sottoprogramma memorizza nel campo frequenza quante volte il suo valore si trova nella prima lista. Se un elemento non è presente nella prima lista, allora deve essere eliminato dalla seconda.

Quesito 6 (5 punti)

Punteggio ottenuto: ... /5

In relazione all'architettura di un sistema di calcolo in termini di hardware e software di sistema, indicare obiettivi, meccanismi e politiche del gestore dei processi, o kernel.

Si preveda di disporre dei sottoprogrammi *lunghezza*, *conta*, *instesta*, *inscoda*, *insordinato*, *esiste*, *del*, *svuotalista*, i cui prototipi (con anche i nomi dei parametri) e funzionalità sono riportati di seguito. I sottoprogrammi qua riportati si riferiscono - per semplicità - al caso di lista per la gestione di dati interi: si immagini di disporre del sottoprogramma equivalente per la gestione di tipi di dati anche diversi, in base alle esigenze.

```
/* restituisce il numero di elementi presenti nella lista h */
int * lunghezza(t_elem * h);
/* restituisce il numero di elementi presenti nella lista h con campo informazione val */
int * conta(t_elem * h, int val);
/* crea un nuovo elemento con campo informazione val e lo inserisce in testa alla lista h, restituendo la testa */
t_elem * instesta(t_elem * h, int val);
/* crea un nuovo elemento di campo informazione val e lo inserisce in coda alla lista h, restituendo la testa */
t_elem * inscoda(t_elem * h, int val);
/* crea un nuovo elemento di campo informazione val e lo inserisce in ordine nella lista h, restituendo la testa */
t_elem * insordinato(t_elem * h, int val);
/* cerca nella lista h un termine con campo informazione val e se esiste restituisce il puntatore a tale termine
 * altrimenti restituisce NULL */
t_term * esiste(t_elem * h, int val);
/* elimina dalla lista h un termine con campo informazione val e restituisce la testa della lista */
t_term * del(t_elem * h, int val);
/* svuota la lista h */
void svuotalista(t_elem * h);
```