

Fondamenti di Informatica - A.A. 2018-2019

Scuola di Ingegneria Industriale e dell'Informazione
Prof.ssa Cristiana Bolchini
Appello del 15/07/2019



POLITECNICO
MILANO 1863

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

Quesito:	1	2	3	4	5	Totale
Peso / Punti Max:	5	4	9	4	8	30
Valutazione in decimi (/10):						

INIZIARE LA SOLUZIONE DI OGNI
ESERCIZIO SU UNA PAGINA NUOVA

RESTITUIRE COMPILATO ANCHE
NEL CASO IN CUI SI RITIRI

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 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.

Quesito 1 [5 punti]

Dati i due numeri $X = 111001110_{2C2}$ e $Y = +4F_{16MS}$ effettuare la conversione in base 2, notazione complemento a 2 (2C2), sul numero minimo di bit per ciascuno degli operandi. Si effettuino quindi le operazioni $X+Y$ e $X-Y$ indicando esplicitamente se si verifica overflow o meno, e motivando la risposta. **Mostrare i passaggi fatti.**

Riportare qua la codifica di X_{2C2} , Y_{2C2} e i risultati finali delle operazioni, utilizzando solo le caselle necessarie (**allineati a destra**) ed indicando se si è verificato overflow.

X_{2C2} :	<input type="text"/>	Y_{2C2} :	<input type="text"/>
		ovf	ovf
$(X + Y)_{2C2}$:	<input type="text"/>	$(X - Y)_{2C2}$:	<input type="text"/>

Quesito 2 [4 punti]

Rappresentare il valore reale $Z = c1540000$ (notazione IEEE754 espresso in base 16) in base 10, notazione modulo e segno. **Mostrare i passaggi fatti.**

Quesito 3 [9 punti]

(7 punti) Scrivere un sottoprogramma `crop` che ricevuta in ingresso una stringa `frase` ed un carattere `ch` restituisce una **nuova** stringa che contiene i caratteri compresi tra la prima e la seconda occorrenza del carattere `ch`, escluso il carattere `ch`. Per esempio, se il sottoprogramma riceve in ingresso `informatica` e `i`, il sottoprogramma restituisce la stringa `nformat`. Nel caso in cui la stringa `frase` non contenga due occorrenze del carattere `ch`, restituisce `NULL`.

(2 punti) Scrivere un programma che acquisisce da riga di comando una stringa ed un carattere e chiama il sottoprogramma `crop` prima descritto e visualizza il risultato dell'elaborazione, quindi termina. Un paio di esecuzioni, da riga di comando, di esempio sono:

```
./ritagliastringa informatica a  
tic  
./ritagliastringa collaudo x  
(null)
```

[nota: quando si esegue una istruzione `printf("%s", variabile)` e la variabile vale `NULL`, a video si vede `(null)`].

Quesito 4 [4 punti]

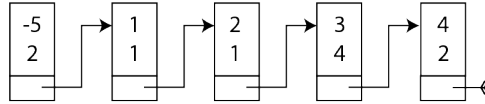
Si scriva un sottoprogramma che ricevuto in ingresso il nome di un file restituisce 1 se il file esiste e si hanno i permessi per accedervi, 0 se il file non esiste, -1 se il file esiste ma non si hanno i permessi per accedervi. Il sottoprogramma non deve in alcun modo alterare il file.

Quesito 5 [8 punti]

(6 punti) Scrivere un sottoprogramma `compactlist` che ricevuta in ingresso una lista per la gestione dei numeri interi ne crei una nuova in cui per ogni valore presente nella lista in ingresso viene memorizzato anche il numero di volte in cui esso compare. La lista creata deve avere gli elementi ordinati in senso crescente rispetto al valore (e non al numero di occorrenze). La lista così creata viene restituita al chiamante. Per esempio, se la lista in ingresso è la seguente:

3 → 3 → 1 → 2 → 4 → 3 → -5 → 3 → -5 → 4 → |

il sottoprogramma restituisce la lista seguente



(2 punti) Definire i due tipi di lista necessari per la realizzazione del sottoprogramma.

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

```

/* inserisce in testa alla lista */
listtype * push(listtype *, int);
/* inserisce in coda alla lista */
listtype * append(listtype *, int);
/* inserisce ordinatamente in lista, in base al valore crescente */
listtype * increasing(listtype *, int);
/* inserisce ordinatamente in lista, in base al valore 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 nella lista che ha il valore indicato, se esiste, NULL altrimenti */
listtype * exists(ilist_t *, int);
/* restituisce il numero di elementi nella lista */
int length(ilist_t *);
/* elimina la lista */
listtype * emptylist(ilist_t *);

```