

Fondamenti di Informatica T-1 (A.A. 2012/2013) - Ingegneria Informatica
Prof.ssa Mello
Prova Parziale d'Esame di Venerdì 14 Giugno 2013 – durata 1h
Totale 12 punti, sufficienza con 7

ESERCIZIO 1 (6 punti)

Date due liste di interi `l1` e `l2` ordinate in ordine crescente e senza ripetizioni, si realizzi una funzione ITERATIVA

```
list find_common(list l1, list l2);
```

che restituisca una nuova lista ordinata in ordine decrescente contenente i soli elementi presenti in entrambe le liste `l1` e `l2`. Ad esempio, se `l1 = [1, 2, 4, 8, 9, 15]` e `l2 = [0, 2, 4, 7, 9]`, la funzione `find_common()` deve restituire la lista `[9, 4, 2]` in quanto solo gli interi 2, 4 e 9 sono presenti sia in `l1` che in `l2`.

La funzione `find_common()` dovrà essere implementata utilizzando le sole primitive dell'ADT lista; ogni altra funzione dovrà essere opportunamente specificata dal candidato. Si realizzi inoltre una semplice funzione `main()` di prova che invochi correttamente la funzione `find_common()` appena creata.

ESERCIZIO 2 (2 punti)

Si consideri la seguente funzione:

```
float ric(float acc, int h){
    if (h == 0)
        return acc;
    else {
        float tmp = acc * h;
        return 1 + ric(tmp, h - 1.0);
    }
}
```

Mostrare la sequenza dei record di attivazione ed il valore di ritorno nel caso in cui la funzione sia invocata con parametri attuali (1.5, 3).

ESERCIZIO 3 (3 punti)

Il seguente programma C compila correttamente? In caso affermativo, quali sono i valori stampati a tempo di esecuzione? (si motivi opportunamente la risposta data)

```
#include <stdio.h>
#include <stdlib.h>

#define DIM 15

int melt(char* s1, char* s2, char *out, int res){
    char diff;

    if (*s1 == '\0' || *s2 == '\0') {
        out[0] = '\0';
        return res;
    } else {
        diff = *s1 > *s2 ? *s1 - *s2 : *s2 - *s1;
        *out = 'a' + diff;
        return melt(s1 + 1, s2 + 1, out + 1, res + diff);
    }
}

int main() {
    char* final;
    int res;

    final = (char*) malloc(sizeof(char) * DIM);
    res = melt("cave", "base", final, 0);
    printf("%s %d\n", final, res);
    return 0;
}
```

ESERCIZIO 4 (1 punto)

Il candidato illustri brevemente cosa è e a cosa serve la funzione C `fopen()`, realizzando un breve esempio di codice che ne faccia uso correttamente.

Soluzioni

ESERCIZIO 1

```
list find_common(list l1, list l2) {
    list tmp;
    list res = emptylist();

    while (!empty(l1)) {
        tmp = l2;
        while (!empty(tmp) && head(tmp) < head(l1))
            tmp = tail(tmp);
        if (!empty(tmp) && head(tmp) == head(l1))
            res = cons(head(l1), res);
        l1 = tail(l1);
    }
    return res;
}

int main(){
    list l1, l2, res;
    l1 = cons(1, cons(2, cons(4, cons(8, cons(9, cons(15, emptylist())))));
    l2 = cons(0, cons(2, cons(4, cons(7, cons(9, emptylist()))));

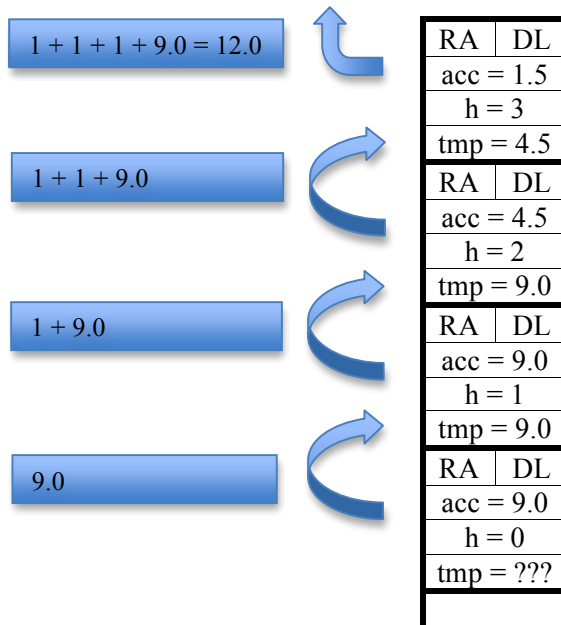
    res = find_common(l1, l2);

    while (!empty(res)) {
        printf("%d ", head(res));
        res = tail(res);
    }
    printf("\n");

    return 0;
}
```

ESERCIZIO 2

La funzione restituisce il valore 12.0.



ESERCIZIO 3

Il programma è corretto sintatticamente e la sua esecuzione produce la stampa:

bada 4

La funzione `main()` alloca spazio sufficiente a contenere 15 char in `final` ed invoca la funzione `melt()` sulle stringhe `s1`, `s2` e `final` con parametro `res` inizializzato a 0.

La funzione ricorsiva `melt()` cicla lungo le stringhe ben formate `s1` e `s2`. Ad ogni invocazione, `melt()` verifica se almeno una delle stringhe puntate da `s1` e `s2` è una stringa ben formata e vuota: in tal caso, inserisce il terminatore di stringa `'\0'` nella stringa puntata da `out` e restituisce il valore accumulato fino a quel momento in `res`. In caso contrario calcola la differenza in valore assoluto `diff` tra il primo carattere di `s1` ed il primo carattere di `s2`, setta il carattere puntato da `*out` ad `'a'` sommato a `diff`, infine invoca `melt()` ricorsivamente avanzando la posizione di `s1`, `s2` e `out` ed accumulando `diff` in `res`.

La funzione `main()` stampa sullo standard output la stringa ben formata presente nell'area di memoria restituita dalla funzione `melt()` e la somma delle differenza in valore assoluto tra i rispettivi caratteri di `s1` ed `s2`.