

Fondamenti di Informatica T-1 (A.A. 2014/2015) - Ingegneria Informatica
Prof.ssa Mello
Prova Parziale d'Esame di Giovedì 11 Giugno 2015 – durata 1h
Totale 12 punti, sufficienza con 7

ESERCIZIO 1 (6 punti)

Data una lista di `char` di nome `car` i cui caratteri sono ordinati in ordine lessicografico, si realizzi una funzione RICORSIVA

```
int consec(list car);
```

che restituisca 1 se la distanza tra elementi consecutivi della lista `car` è sempre inferiore o uguale a 3, 0 altrimenti. Ad esempio, se `car = ['a', 'c', 'e', 'f']` la funzione `consec` restituisce 1, se `car = ['a', 'c', 'm', 'n']` la funzione `consec` restituisce 0 in quanto la distanza tra 'c' e 'm' è superiore a 3.

La funzione `consec()` dovrà essere implementata utilizzando le 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 `consec()` creata.

ESERCIZIO 2 (2 punti)

Un elaboratore rappresenta i numeri interi su 8 bit tramite la notazione in complemento a 2. Indicare come viene svolta la seguente operazione aritmetica calcolandone il risultato secondo la rappresentazione binaria in complemento a 2 (si trasli anche il risultato in decimale per verificare la correttezza dell'operazione):

$$-57 + 67$$

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>
#include <string.h>

char* modify(char* s, int* v){
    int i=0;
    char * res;

    for( ; s[i]!='\0'; i++);

    res = (char*)malloc(sizeof(char)*(i+1));

    for(i=0; s[i]!='\0'; i++){
        if( i % 2 == 0) res[i] = *(s+i) + v[i];
        else res[i] = *(s+i) - v[i];
    }
    *(res+i) = '\0';

    return res;
}

int main(){
    char *stringa = "ciao";
    char *post;
    int push[4] = {-1, 0, +1, +2};

    post = modify(stringa, push);

    printf("pre %s\n", stringa);
    printf("post %s\n", post);

    return 0;
}
```

ESERCIZIO 4 (1 punto)

Il candidato illustri brevemente cosa si intende per **cast** esplicito, fornendo un esempio concreto di cast.

Soluzioni

ESERCIZIO 1

```
int consec(list l){
    int diff;
    if( empty(l) ) return 1;
    else if( empty(tail(l)) ) return 1;
    else{
        diff = head(tail(l)) - head(l);
        if( diff > 3 ) return 0;
        else return ( consec(tail(l)) ) ;
    }
}

int main(){
    int res;

    list l = cons('a',cons('c',cons('e',cons('f',emptylist()))));
    res = consec(l);
    printf("%d\n", res);

    l = cons('a',cons('c',cons('m',cons('n',emptylist()))));
    res = consec(l);
    printf("%d\n", res);

    return 0;
}
```

ESERCIZIO 2

$$- 57 + 67 = 10$$

$$+ 57 \rightarrow 00111001 = 32 + 16 + 8 + 1$$

$$11000110$$

$$- 57 \rightarrow 11000111 = -128 + 64 + 4 + 2 + 1$$

$$+ 67 \rightarrow 01000011 = 64 + 2 + 1$$

$$+ 10 \rightarrow 00001010 = 8 + 2$$

ESERCIZIO 3

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

```
pre ciao
post bibm
```

La funzione `main()` invoca la funzione `modify()` con parametri di ingresso una stringa ben formata e una array di interi.

La funzione `modify()` utilizza un ciclo `for` con corpo vuoto per contare il numero di caratteri presenti nella stringa ben formata `s` (terminatore di stringa escluso), ovvero 4. In seguito alloca dinamicamente spazio sufficiente per contenere 5 caratteri. Infine la funzione `modify()` copia nello spazio allocato dinamicamente i caratteri presenti nella stringa ben formata `s`, shiftandoli di un numero di posizioni corrispondente al valore dell'*i*-esimo elemento dell'array di interi `v`, considerandone il valore positivo in caso di elemento di indice pari e il valore negativo in caso di elemento di indice dispari. Terminato il secondo ciclo `for` la funzione `modify()` inserisce il terminatore di stringa in fondo all'area di memoria allocata dinamicamente e restituisce al chiamante un riferimento a tale area di memoria.

La funzione `main()` stampa sullo standard output la stringa originaria ("ciao") e la stringa restituita ("bibm").