

Fondamenti di Informatica T-1 (A.A. 2017/2018) - Ingegneria Informatica
Prof.ssa Mello
Prova Scritta - 15 Giugno 2018 – durata 1h
Totale 12 punti, sufficienza con 7

ESERCIZIO 1 (6 punti)

Si scriva una funzione RICORSIVA `onlyFollowing`:

```
list onlyFollowing(list list1, char car, int N);
```

che data una lista di caratteri `list1` e un carattere `car`, restituisca una nuova lista contenente i soli caratteri di `list1` che hanno rispetto a `car` una distanza in valore assoluto minore o uguale a `N`.

Ad esempio se `list1 = ['a', 'v', 'r', 'c', 'm', 'd']`, `car = 'c'` e `N = 5`, la funzione `onlyFollowing()` deve restituire la lista `['a', 'c', 'd']`.

La funzione dovrà essere implementata utilizzando le sole primitive dell'ADT lista, includendo "list.h". Ogni altra funzione dovrà essere opportunamente specificata dal candidato. Si realizzi inoltre una semplice funzione `main()` di prova che invochi correttamente la funzione `onlyFollowing()`.

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):

5 – 12

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>

int N=3;

void change(char* vec, int i){
    int N=0;
    vec[i%4] = 'E'+1;
    N++;
}

void sub(int *i){
    (*i)--;
    N++;
}

int main(){
    int i;
    char s[] = "LIFO";
    for (i=6; i > 1; i-=2){
        if(i/N > 0){
            change(s,i);
        }else{
            sub(&i);
        }
    }
    printf("N=%d\ni=%d\ns \n",N,i,s);
    return 0;
}
```

ESERCIZIO 4 (1 punto)

Si illustri il concetto di allocazione dinamica di un blocco di memoria (`malloc`) e se ne fornisca un breve esempio

Soluzioni

ESERCIZIO 1

```
#include <stdio.h>
#include "list.h"

int assoluto(int N){
    if (N<0) return -N;
    else return N;
}

list onlyFollowing(list list1, char c, int N){
    if (empty(list1))
        return list1;
    else {
        if ((assoluto(head(list1)- c)) <=N ){
            return cons(head(list1), onlyFollowing(tail(list1),c, N));
        }else{
            return onlyFollowing(tail(list1),c,N);
        }
    }
}

int main(){
    list list1, result;
    list1 = cons('a', cons('v', cons('r',
        cons('c', cons('m', cons('d', emptylist())))));
    result = onlyFollowing(list1, 'c', 5);
    while( ! empty(result) ){
        printf("%c\n", head(result));
        result = tail(result);
    }
}
```

ESERCIZIO 2

5 - 12 = -7
5 → 00000101
12 = 8 + 4 → 00001100
- 12 → 11110100
5 - 12 → 11111001
11111001 = -128+64+32+16+8+1 = -7

ESERCIZIO 3

L'output prodotto è

```
N=4
```

```
i=-1
```

```
FIFO
```

Il programma inizializza una variabile globale `N=3`. Il `main` crea una stringa `s="LIFO\0"` e poi esegue il ciclo `for` con `i=6`. Poiché `i/N=2`, viene eseguita la funzione `change` con parametri `vec="LIFO\0"` (passaggio per riferimento) e `i=6` (per valore).

All'interno della funzione, viene dichiarata la variabile `N` (locale a `change`) che assume valore `0`. Poiché `i%4=2`, la funzione associa a `s[2]` il carattere `'F'` (successivo a `'E'` in ordine alfabetico) lasciando di fatto immutata la stringa `s`. `N++` modifica la variabile locale `N` che diventa `1`.

Al ciclo `for` successivo `i=4`. Poiché `i/N=4/3=1`, viene eseguita la funzione `change` con parametri `vec="LIFO\0"` e `i=4`. L'istruzione `vec[i%4] = 'E'+1` modifica il primo elemento di `s` che diventa così `"FIFO\0"`.

Alla successiva esecuzione del `for` `i=2`, per cui viene eseguita la funzione `sub`. Poiché il parametro `i` viene passato per riferimento, la funzione decrementa la variabile `i` del `main` che diviene `1`, mentre il valore della variabile globale `N` viene incrementato (`N=4`).

L'esecuzione successiva del ciclo porta `i` al valore `-1`. La condizione `i>1` non è più soddisfatta, per cui il ciclo termina e viene eseguita la `printf`.