

Fondamenti di Informatica T-1 (A.A. 2012/2013) - Ingegneria Informatica
Prof.ssa Mello
Prova Parziale d'Esame di Venerdì 15 Febbraio 2013 – durata 1h
Totale 12 punti, sufficienza con 7

Compito A

ESERCIZIO 1 (6 punti)

Data una lista di stringhe ben formate `l`, si realizzi una funzione RICORSIVA

```
list filterString(list l);
```

che restituisca una nuova lista contenente gli elementi presenti nella lista `l` che abbiano almeno 2 caratteri alfabetici minuscoli. Ad esempio, se `l = ["AbA", "aBa", "abc", "ABC", "!aiuOLA*"]`, la funzione `filterString()` deve restituire la lista `["aBa", "abc", "!aiuOLA*"]` in quanto "aBa" ha due caratteri alfabetici minuscoli, "abc" tre e "!aiuOLA*" tre.

A tal scopo si implementi e si faccia uso della funzione ITERATIVA

```
int contaMin(char *str);
```

che, data in ingresso una stringa ben formata, restituisce il numero di caratteri alfabetici minuscoli presenti in tale stringa.

Le funzioni `filterString()` e `contaMin()` dovranno essere implementate 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 `filterString()` 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):

+ 29 – 53

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 countA(char* stringa){
    if( *stringa == '\0' ){
        return 0;
    }
    else{
        return (*stringa - 'b') + countA(&stringa[1]);
    }
}

int main() {
    char* str = "abcb";
    int i = 0, res;
    char* new;

    new = (char*) malloc(sizeof(char)*10);

    while( *str != '\0' ){
        new[i] = *str + 1;
        i++;
        str++;
    }
    new[i] = '\0';

    res = countA(new);

    printf("%d %s\n",res, new);

    return 0;
}
```

ESERCIZIO 4 (1 punto)

Si descriva cosa succede durante il passaggio per riferimento di un parametro a una funzione utilizzando il linguaggio C; in particolare si analizzino le differenze principali tra il passaggio di un array e di una struct.

Soluzioni

ESERCIZIO 1

```
int contaMin(char* str){
    int min = 0;
    while( *str != '\0' ){
        if( *str >= 'a' && *str <= 'z' ){
            min ++;
        }
        str++;
    }
    return min;
}

list filterStrings(list l){
    int min;
    if( empty(l) ){
        return l;
    }
    else {
        min = contaMin(head(l));
        if( min >= 2 ){
            return cons(head(l),filterStrings(tail(l)));
        }
        else {
            return filterStrings(tail(l));
        }
    }
}

int main(){
    list l, res;

    l = cons("AbA", cons("aBa", cons("abc", cons("ABC", cons("!aiuOLA*",
emptylist())))));

    printf("filterStrings\n");
    res = filterStrings(l);

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

    return 0;
}
```

ESERCIZIO 2

+ 29 - 53 = - 24
+ 29 ==> 00011101
+ 53 ==> 00110101
 11001010
- 53 ==> 11001011
- 24 ==> 11101000 = -128 + 64 + 32 + 8

ESERCIZIO 3

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

4 bcdc

La funzione `main()` alloca spazio sufficiente a contenere 10 char e poi inserisce in tale spazio di memoria i caratteri presenti nella stringa ben formata `str` shiftati di uno. Successivamente la funzione `main()` invoca la funzione `countA()`.

La funzione ricorsiva `countA()` restituisce 0 se la stringa passata come parametro in ingresso è vuota. In caso contrario, la funzione `countA()` restituisce la distanza tra il primo carattere della stringa `str` e il char 'b' sommato a ciò che viene restituito dalla chiamata ricorsiva alla funzione stessa passando come parametro d'ingresso l'indirizzo del secondo carattere della stringa `str`.

La funzione `main()` stampa sullo standard output il valore restituito dalla funzione `countA()` e la stringa ben formata presente nell'area di memoria allocata dinamicamente.