

Consorzio NETTUNO – Ing. Informatica, Elettronica, Comunicazioni
Fondamenti di Informatica 1 – 20/1/2007 – Andrea Sterbini

Esercizio 1 (stringhe)

Si scriva la funzione C che riceve come argomenti: una stringa **testo** ed una stringa **cerca**, e che torna vero (1) se la stringa cerca è presente due volte nella stringa testo, la prima dritta e la seconda rovesciata, altrimenti torna falso (0).

Esempio: se testo = “**la** vecchia con la borsa **salta** il fosso senza rincorsa”
cerca = “orsa”

allora il risultato è 0 (la stringa “orsa” vi appare ma non c'e' la stringa rovescia “asro”)
mentre sarebbe 1 se la stringa da cercare fosse “**la**” (infatti c'è la stringa a rovescio “**al**”)

Esercizio 2 (vettori)

Si scriva un programma C che legge da standard input una successione di interi. Si assuma che i valori letti siano sempre contenuti tra **-10** e **+110** (inclusi) e che si smetta di leggere quando si riceve un valore fuori da tale range. Il programma deve costruire un istogramma dei valori, ovvero un vettore in cui si conta, per ciascun valore possibile tra -10 e +110, quante volte quel valore è apparso nell'input.

Infine il programma stampa l'istogramma, stampando via via, su una riga per volta:

- il **valore** corrispondente, poi uno **spazio**, poi un numero di **asterischi** pari al numero di valori letti.

Esempio: se i valori letti fossero: **-9 -7 -9 -9 -7 -9 -10 -8 -7 -8 200** si stamperebbe

```
-10 *  
-9  ****  
-8  **  
-7  ***
```

eccetera fino alla riga del valore 110 ...

Esercizio 3 (matrici)

Si scriva la funzione C che riceve come argomenti:

- una matrice **M** bidimensionale di 27 righe e 39 colonne
- quattro interi **minX**, **minY**, **maxX** e **maxY** che indicano le coordinate minime e massime di un rettangolo di elementi di M

La funzione calcola e torna la media dei valori contenuti nel rettangolo individuato.

1	2	9	0	6	5	0	9
8	3	4	8	9	5	7	8
0	7	5	3	6	4	6	5
8	3	5	4	9	2	0	8
8	8	0	9	4	2	4	9
6	9	4	3	5	5	8	7
8	9	9	8	0	7	0	7
5	0	7	5	3	4	9	0

minX=1, minY=1,
maxX=3, maxY=5

Media: 5

Esercizio 4 (ricorsione)

Si scriva la funzione ricorsiva che calcola F(x, y, z), definita come:

$$\begin{aligned} F(x, y, z) &= 1 + F(x-1, y, z) && \text{se } x > 0 \\ F(x, y, z) &= 2 + F(x, y-1, z) && \text{se } y > 0 \\ F(x, y, z) &= 5 + F(x, y, z-1) && \text{se } z > 0 \\ F(x, y, z) &= 42 && \text{altrimenti} \end{aligned}$$

(ha la precedenza la formula che corrisponde alla prima condizione che è vera, nell'ordine dato qua sopra)

- Si descriva nel dettaglio la sequenza delle chiamate ricorsive della funzione F(2, 3, 1) mostrando sia le **chiamate** della funzione con i relativi argomenti che i **ritorni** della funzione con i relativi valori.
- Si indichi a quale funzione algebrica viene calcolata dalla funzione F(x, y, z).

Esercizio 5 (Algebra di Boole)

Si applichino le regole di De Morgan alla espressione booleana che segue

$$\underline{a} \underline{b} \underline{c} + (\underline{a} \underline{b} \underline{c} + a \underline{b}) (\underline{c} + \underline{b}) \underline{c}$$

NOTA: le espressioni sottolineate vanno intese come NEGATE

Soluzioni

Esercizio 1

```
#define TRUE 1
#define FALSE 0
#include <stdlib.h>
int cercaDrittoRovescio(char * testo, char * cerca) {
    int i, j, found;
    int lenT = strlen(testo);           // lunghezza del testo
    int lenC = strlen(cerca);          // lunghezza della stringa da cercare
    // cerco la prima occorrenza della stringa cerca
    for (i=0 ; i<lenT-lenC ; i++) {     // per ogni posizione possibile nel testo
        found = TRUE;                  // flag che indica "trovato"
        for (j=0 ; j<lenC ; j++) {     // scandisco tutta la stringa da cercare
            if (cerca[j] != testo[i+j]) // se il carattere è diverso
                found = FALSE;         // ricordo che non va bene
            break;                      // e passo alla prossima i
        }
        if (found) break;              // ma se l'ho trovata esco dal ciclo
    }
    // torno FALSE se non c'e' nemmeno la prima stringa
    if (!found) return FALSE;
    // cerco la seconda occorrenza della stringa cerca a rovescio
    for (i=i+lenC ; i<lenT-lenC ; i++) { // per ogni pos. possibile nel seguito
        found = TRUE;                  // flag che indica "trovato"
        for (j=0 ; j<lenC ; j++) {     // scandisco tutta la stringa da cercare
            if (cerca[lenC-j] != testo[i+j]) // se il carattere è diverso
                found = FALSE;         // ricordo che non va bene
            break;                      // e passo alla prossima i
        }
        if (found) break;              // ma se l'ho trovata esco dal ciclo
    }
    return found;
}
```

Esercizio 2

```
#define START -10
#define END 110
// calcolo il numero di valori necessari (110-(-10)+1) = 121
#define NUMVAL (END-START+1)
int main() {
    int i, j, valore;
    int istogramma[NUMVAL] = { 0 };    // inizializzo l'istogramma con zeri
    scanf("%d", &valore);              // leggo il primo valore
    while (valore >= START && valore <= END) { // smetto se è fuori range
        istogramma[valore-START] ++;    // conto il valore letto
        scanf("%d", &valore);          // ne leggo un altro
    }
    // stampo l'istogramma
    for (i=0 ; i< NUMVAL ; i++) {       // per ogni elemento
        printf("\n%d ", i+START);       // stampo il numero del valore
        for (j=0 ; j<istogramma[i] ; j++) // e gli asterischi
            printf("*");
    }
    return 0;
}
```

Esercizio 3

```
#define DIMX 39
#define DIMY 27
Float mediaDelRettangolo( int minX, int maxX, int minY, int maxY, int M[DIMX][DIMY]) {
    int quanti = 0;
    int somma = 0;
    int i, j;
    for (i=minX ; i<=maxX && i < DIMX ; i++)
        for (j=minY ; j <= maxY && j < DIMY ; j++) {
            somma += M[i][j];
            quanti ++;
        }
    return somma/quanti;
}
```

Esercizio 4

```
int F (int x, int y, int z) {
    if (x>0)
        return 1 + F(x-1,y,z);
    if (x>0)
        return 2 + F(x,y-1,z);
    if (x>0)
        return 5 + F(x,y,z-1);
    return 42;
}
```

La successione delle chiamate e dei ritorni per F(2,3,1) è:

```
F(2,3,1)
  F(1,3,1) + 1
    F(0,3,1) + 1
      F(0,2,1) + 2
        F(0,1,1) + 2
          F(0,0,1) + 2
            F(0,0,0) + 5
              F(0,0,0)
                torna 42
              torna 42+5=47
            torna 47 + 2 = 49
          torna 49 + 2 = 51
        torna 51 + 2 = 53
      torna 53 + 1 = 54
    torna 54 + 1 = 55
```

Da cui si vede che la funzione F(x,y,z) calcola la espressione $42 + 5z + 2y + x$

Esercizio 5

Le leggi di DeMorgan dicono che date due espressioni booleane A e B, la negazione della somma e del prodotto è (rispettivamente) uguale al prodotto ed alla somma di A e B negati, ovvero

$$\begin{aligned} \overline{A+B} &= \overline{A} * \overline{B} \\ \overline{A*B} &= \overline{A} + \overline{B} \end{aligned}$$

quindi l'espressione diventa (esplicitando i prodotti)

$$\underline{a} \underline{b} \underline{c} + (\underline{a} \underline{b} \underline{c} + \underline{a} \underline{b}) (\underline{c} + \underline{b}) \underline{c} = (\underline{a} + \underline{b}) * \underline{c} + (\underline{a} + \underline{b} + \underline{c} + \underline{a} * \underline{b}) * (\underline{c} * \underline{b}) * \underline{c}$$