

LABORATORIO FONDAMENTI DI INFORMATICA - 28-29 OTTOBRE 2013

Esercizio 1

Scrivere un programma che chieda continuamente un numero intero positivo n all'utente fino a quando viene inserito il valore 0. Il programma deve stampare sullo schermo tanti asterischi "*" quanti sono indicati dal numero n.

Infine il programma deve stampare a video quanti asterischi sono stati stampati in totale.

```
numero di '*'? 5
*****
numero di '*'? 3
***
numero di '*'? 0
stampati 8 '*'
```

Esercizio 2

La tavola pitagorica è una matrice bidimensionale i cui elementi sono numeri interi calcolati moltiplicando l'indice della riga con l'indice della colonna alle quali l'elemento appartiene.

Scrivere un programma che stampi a video una tavola pitagorica di dimensioni 10 x 10.

Gli elementi della matrice che risultano essere multipli sia di 5 che di 7 (contemporaneamente, ad esempio 35, 70, ...) devono essere evidenziati affiancandoli con un asterisco (*).

NOTE:

- non utilizzare i vettori o le matrici (stampare direttamente a video gli elementi)
- definire la dimensione 10 con una costante DIM: `#define DIM 10`
- per stampare un numero occupando 4 "spazi": `printf("...%4d...", i);`

Esempio di tavola pitagorica 4 x 4:

	1	2	3	4
1	1	2	3	4
2	2	4	6	8
3	3	6	9	12
4	4	8	12	16

Tavola pitagorica 10 x 10:

	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35*	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35*	42	49	56	63	70*
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70*	80	90	100

Esercizio 3

Data una costante intera DIM definita uguale a 7, scrivere un programma che visualizzi una figura quadrata composta da DIM righe e DIM colonne come quella indicata:

```
*+++++
-*++++
--*+++
---*++
----*+
-----*
-----*
```

Lungo la diagonale principale compaiono degli asterischi, al di sotto della diagonale compaiono dei "segni meno -" e al di sopra compaiono dei "segni più +".

Esercizio 4

Scrivere un programma che recuperi uno per volta dei numeri interi pseudo-casuali da 0 a 99 tramite l'uso delle seguenti righe di codice:

```
#include<stdlib.h>
int num;
srand(time(NULL)); //la serie di numeri generati cambia in base al numero di secondi
                    trascorsi dal 1 gennaio 1970 (pseudo-casualità)
...
num = rand() % 100; //istruzione da inserire in un ciclo - num assume valori da 0 a 99
```

I numeri vanno continuamente recuperati fino a quando si verifica almeno una delle seguenti condizioni:

- 1) la quantità totale di numeri interi forniti supera 10 e la quantità di numeri pari forniti è maggiore di almeno due unità rispetto alla quantità di numeri dispari.
- 2) la quantità totale di numeri interi forniti raggiunge 20.

Per verificare che il programma funzioni correttamente, si consiglia di far stampare al programma quanti numeri sono stati forniti e inoltre quanti di questi sono pari e quanti sono dispari.

Esempio in cui il programma si ferma per la condizione 1):

```
Ricevuto 44, in totale ho ricevuto 1 numero di cui 1 pari e 0 dispari
...
Ricevuto 36, in totale ho ricevuto 16 numeri di cui 9 pari e 7 dispari
Fine
```

Esempio di esecuzione in cui il programma si ferma per la condizione 2):

```
Ricevuto 79, in totale ho ricevuto 1 numero di cui 0 pari e 1 dispari
...
Ricevuto 71, in totale ho ricevuto 20 numeri di cui 9 pari e 11 dispari
Fine
```

Esercizio 5

Data la costante SUPERIORE definita uguale a 10, scrivere un programma che per ogni numero X compreso nell'intervallo 1..SUPERIORE (estremi compresi) stampi la somma dei 7 numeri successivi a X che hanno la stessa parità di X (ovvero se X è pari occorre sommare solo i numeri pari, se X è dispari occorre sommare solo i numeri dispari)

```
X= 1  dispari  somma = 3 + 5 + 7 = 15
X= 2    pari   somma = 4 + 6 + 8 = 18
X= 3  dispari  somma = 5 + 7 + 9 = 21
X= 4    pari   somma = 6 + 8 + 10 = 24
...
X=10    pari   somma = 12 + 14 + 16 = 42
```

Esercizio 6

Data la costante INFERIORE definita uguale a 50, scrivere un programma che per ogni numero X compreso nell'intervallo [$<\text{inferiore}>..70$] (estremi compresi) stampi i seguenti risultati:

- per i numeri X NON multipli di 3 e NON multipli di 7, è necessario stampare come risultato "0"
- per i numeri X multipli di 3 e di 7 (di entrambi) è necessario stampare come risultato "-1"
- per i rimanenti numeri X (ovvero quelli multipli di 3 o di 7 ma non di entrambi), stampare come risultato la somma dei numeri compresi nell'intervallo [(doppio di X) .. 150] estremi compresi, aventi la stessa caratteristica di X (ovvero, se X è multiplo di 3, devono essere sommati i multipli di 3; invece se X è multiplo di 7 devono essere sommati i multipli di 7).

```
50: ne' 3 ne' 7 -> 0
51: solo 3 -> [102..150] 102+105+108+111+114+117+120+123+126+129+132+135+138
+141+144+147+150=2142
...
56: solo 7 -> [112..150] 112+119+126+133+140+147=777
...
63: sia 3 che 7 -> -1
...
```