Programmazione
I compitino (Vers. B)

28 novembre 2012

Cognome .... Nome ....
Matricola ....

Note.

- In alcune delle seguenti domande verrà usato il vostro numero di matricola. Scrivete il vostro numero nel seguente schema, una cifra per ogni spazio:

(A)	(B)	(C)	(D)	(E)	(F)

  Negli esercizi, useremo le lettere (A), (B), (C)... per riferirci alle corrispondenti cifre del numero di matricola. Ad esempio, "il numero $(D)(E)(F)$" è il numero costituito dalle ultime tre cifre del vostro numero di matricola.

- Ove non sia esplicitamente indicato, per "programma" si intende semplicemente il corpo della funzione main.

Rispondete alle seguenti domande.

1. Assumete che x e y siano variabili **int**, e che il loro valore *prima di ciascuno dei seguenti assegnamenti* sia rispettivamente $(A)(F)$ e $(E)(D)$. Dite quale sarà il loro valore *dopo* gli assegnamenti indicati:

Assegnamento	x	y
x=(y=x/2)		
x=y%7? x : y++		
x+=(5+y)/3		
y−=−−x		

2. Considerate il seguente programma C:

```
int x, y, z, c;
scanf( "%d", &x );
z = 1;
c = 0;
while ( z <= x ) {
    c++;
    z *= 2;
}
printf( "%d\n", z / 2 );
```

- Cosa stampa il programma quando $x = (A)(E)$? ....
- Cosa stampa il programma quando $x = (E)(D)$? ....

3. Sia data la seguente dichiarazione di variabili:

```
int x = (A)(F);
int y = (D)(E);
```

Dire qual è il valore di verità delle seguenti espressioni (indicate *vero* se l'espressione ha un valore numerico diverso da zero, *falso* se l'espressione ha un valore numerico uguale a zero) e qual è il contenuto delle variabili x e y dopo la valutazione delle espressioni:

Espressione	**Valore di verità**	x	y
`y>x && ++y>x++`			
`y>2*x || x++>(y-=3)`			

4. Il seguente programma dovrebbe leggere una stringa e stamparne i soli caratteri numerici. Il programma contiene degli errori; segnateli con un cerchietto.

```
#include "stdio.h"
#include "string.h"
#define MAX_STR 500;

int main( void ) {
 int i, n;
 char x[ MAX_STR ];

 gets( x );
 strlen( x );
 for ( i = 1; i < n; i++ )
  if ( x[ i ] >= '0' || x[ i ] <= '9')
   printf( "%c", x );
 printf( "\n" );
 return 0;
}
```

5. Scrivete un programma che chieda ripetutamente all'utente di inserire degli interi (che si assume siano tutti compresi fra 10 e 99), e che termini quando l'utente inserisce il valore 0. Al termine dell'inserimento, il programma deve stampare la somma delle cifre delle decine. Ad esempio, se l'utente inserisce *51 33 17 50 40 41 0* il programma deve stampare 22 (che è $5 + 3 + 1 + 5 + 4 + 4$).

6. Scrivete un programma che, su un certo input intero (diciamo t), produca un output di t righe, come nell'esempio seguente:

```
Intero? 10
J J J J J J J J J J
I I I I I I I I I I
H H H H H H H H H H
G G G G G G G G G G
F F F F F F F F F F
E E E E E E E E E E
D D D D D D D D D D
C C C C C C C C C C
B B B B B B B B B B
A A A A A A A A A A
```

7. Scrivete un programma che chieda ripetutamente all'utente di inserire degli interi (al massimo 100), e che termini quando l'utente inserisce il valore 0. Al termine dell'inserimento, il programma deve stampare i valori introdotti eccettuato quelli che sono seguiti da un valore maggiore. Ad esempio, se l'utente inserisce *61 61 300 301 61 36 36 34 47 0* il programma deve stampare *61 301 61 36 36 47*.

8. Scrivete un programma che legga un numero intero $n \geq 100$, seguito da n altri interi, e poi ne stampi, per ciascuno, altrettanti spazi e un asterisco:

```
Quanti valori?  6
 3  5  1  3  6  6
   *
     *
 *
   *
      *
      *
```

9. Scrivete un programma che, letta una stringa (di al massimo 500 caratteri), la ristampi saltando le lettere minuscole. Ad esempio:

```
Stringa? QueSTa STRInga sarebbe piUTTosto LUNGA!
QST STRI  UTT LUNGA!
```