

Programmazione

I compito (Preparazione)

1 dicembre 2010

Cognome **Carfagna** Nome **Mara**
Matricola **696969**

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 17 e 25. Dite quale sarà il loro valore *dopo* gli assegnamenti indicati:

| Assegnamento | x | y |
|---|-----------|-----------|
| $x=(x++>0? --y : y++)$ | 24 | 24 |
| $x=(x\%y)+1$ | 18 | 25 |
| $x=(y++)/x$ | 1 | 26 |
| $y=(\mathbf{int})(1+(\mathbf{float})y/x)$ | 17 | 2 |

2. Considerate il seguente programma C:

```
int x, y, z, c;
scanf( "%d", &x );
c = 0;
while ( x > 0 ) {
    if ( x % 10 == 0 )
        c++;
    x /= 10;
}
printf( "%d\n", c );
```

- Cosa stampa il programma quando $x = 1010$? **2**
- Cosa stampa il programma quando $x = 2001001$? **4**

3. Sia data la seguente dichiarazione di variabili:

```
int x = 13;
int y = 14;
```

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 |
|-----------------------------|------------------|-----------|-----------|
| $y > ++x \ \&\& \ y == x++$ | vero | 15 | 14 |
| $x < y \ \&\& \ y == x++$ | falso | 14 | 14 |

4. Il seguente programma dovrebbe leggere una stringa e calcolare il numero di spazi in essa contenuti. Il programma contiene degli errori; segnatele con un cerchietto.

```
#include <stdio.h>
#include <string.h>
#define MAX_STR 500

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

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

Versione corretta:

```
#include <stdio.h>
#include <string.h>
#define MAX_STR 500

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

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

5. Scrivete un programma che permetta all'utente di inserire dei caratteri terminati dall'a-capo. Al termine dell'inserimento, il programma deve stampare il numero di caratteri alfabetici minuscoli. Ad esempio, se l'utente inserisce *6 cos3tto a NON s8linearlo* il programma deve stampare 16.

```
#include <stdio.h>

int main( void ) {
    int c;
    char x;

    c = 0;
    while ( ( x = getchar() ) != '\n' )
        if ( x >= 'a' && x <= 'z' )
            c++;
    printf( "%d\n", c );
    return 0;
}
```

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

```
Intero? 10
0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1
```

```
#include <stdio.h>

int main( void ) {
    int t, i, j;

    printf( "Intero?_" );
    scanf( "%d", &t );
    for ( i = 0; i < t; i++ ) {
        for ( j = 0; j < t; j++ )
            printf( "%d_", i % 2 );
        printf( "\n" );
    }
    return 0;
}
```

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 l'elenco dei valori introdotti eccettuato i multipli di 3. Ad esempio, se l'utente inserisce *61 300 301 36 34 49 47 0* il programma deve stampare *61 301 34 49 47*.

```
#include <stdio.h>

int main( void ) {
    int x[ 100 ], y, n, i;

    n = 0;
    do {
        scanf( "%d", &y );
        if ( y != 0 )
            x[ n++ ] = y;
    } while ( y != 0 );
    for ( i = 0; i < n; i++ )
        if ( x[ i ] % 3 != 0 )
            printf( "%d_", x[ i ] );
    printf( "\n" );
    return 0;
}
```

8. Scrivete un programma che legga un numero intero $n \leq 100$, seguito da n altri interi, e poi ne stampi, per ciascuno, la radice quadrata approssimata per difetto; per farlo *non si deve utilizzare nessuna funzione della libreria matematica*:

```
Quanti valori? 6
153 8123 123122 419 444 6
12
90
350
20
21
2
```

```
#include <stdio.h>

int main( void ) {
    int x[ 100 ], y, n, i;

    printf( "Quanti_valori?_" );
    scanf( "%d", &n );
    for ( i = 0; i < n; i++ )
        scanf( "%d", &x[ i ] );
    for ( i = 0; i < n; i++ ) {
        y = 0;
        while ( y * y <= x[ i ] )
            y++;
        y--;
        printf( "%d\n", y );
    }
    return 0;
}
```

9. Scrivete un programma che, letta una stringa (di al massimo 500 caratteri), la ristampi tutta eccettuati il primo e l'ultimo carattere. Ad esempio:

Stringa? **questa stringa sarebbe piuttosto lunga**
uesta stringa sarebbe piuttosto lung

```
#include <stdio.h>
#include <string.h>
#define MAX 500

int main( void ) {
    int i, n;
    char s[ MAX + 1 ];

    printf( "Stringa?_" );
    gets( s );
    n = strlen( s );
    for ( i = 1; i < n - 1; i++ )
        printf( "%c", s[ i ] );
    printf( "\n" );
    return 0;
}
```