Fondamenti di Architetture e Programmazione
II Compitino

Esercizi di preparazione

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

Rispondete alle seguenti domande[1]:

1. Assumete che x e y siano variabili **int**, e che il loro valore *prima di ciascuno dei seguenti assegnamenti* sia rispettivamente 4 e 10. Dite quale sarà il loro valore *dopo* gli assegnamenti indicati:

Assegnamento	x	y
y=x++	**5**	**4**
x=(y=x)	**4**	**4**
y=(x>=4)? x+1 : x−1	**4**	**5**
x=y + (**int**)3.6	**13**	**10**
x=(y=x)+y	**8**	**4**
y+=x	**4**	**14**

2. Il seguente programma permette a un utente di introdurre numeri interi, usando il valore speciale 0 per terminare. Alla fine, il programma deve stampare quanti dei valori introdotti hanno il quintuplo divisibile per 15. Nei conteggi è compreso lo 0 finale. Completate le parti mancanti (indicate con dei puntini):

[1] Nei programmi, assumeremo sempre che in e out siano due variabili di classe ConsoleInputManager e ConsoleOutputManager (rispettivamente), già dichiarate e inizializzate.

```
int x, div = 0;

do {

    x = in.readInt();

    if ( 5 * x % 15 == 0 )

        div++;

} while ( x != 0 );

out.println( div );
```

3. Il seguente programma, dopo aver letto una stringa, deve stampare quante sono le lettere maiuscole nella stringa. Per esempio, se la stringa è *cA3Ne* deve stampare 2. Completate le parti mancanti (indicate con dei puntini):

```
String s;

int lung, i, maiuscole;

s = in.readLine();

lung = s.length();

maiuscole = 0;

for ( i = 0; i < lung; i++ )

   if ( Character.isUppercase( s.charAt( i ) ) )

        maiuscole++;

out.println( maiuscole );
```

4. Considerate il seguente programma Java?

```
int x, y;
x = in.readInt();
y = in.readInt();
while ( x >= y )
  x=x-y;
out.println(x);
```

   - Cosa stampa il programma quando $x = 15$ e $y = 4$? **3**
   - Cosa succede se $x = 5$ e $y = 0$? **Non termina**

5. Supponete di avere dichiarato e inizializzato le seguenti variabili:

```
int x = 1, y = 2;
long z = 3;
double d = 3.9;
float f = 2.0F;
```

Per ciascuno dei seguenti assegnamenti, indicate se l'assegnamento produce un errore di compilazione, e in caso negativo dite che valore viene assegnato alla variabile:

- `x = x + (int)d`: **4**
- `f =  y`: **2**
- `d = f + x`: **3**
- `f = d + x`: **Errore di compilazione**
- `d = (int)f`: **2**

6. Sia data la seguente definizione di variabili:

```
int x = 3;
int y = 4;
```

Dire qual è il valore di verità delle seguenti espressioni booleane e qual è il contenuto delle variabili x e y dopo la valutazione delle espressioni:

Espressione	**Valore di verità**	x	y
`(y=3*x)%2==0 && (x++>0)`	**false**	**3**	**9**
`(y=x)>2 | (x++>3)`	**true**	**4**	**3**

7. La classe Character contiene un metodo statico di nome isWhitespace che ha come solo argomento un **char** e restituisce un **boolean** che è vero se il carattere passato come argomento è un whitespace (uno spazio, un TAB ecc.) e falso altrimenti; il prototipo del metodo è

```
public static boolean isWhitespace(char ch)
```

Supponete che c sia una variabile di tipo **char** e di voler determinare se il suo contenuto è un whitespace, mettendo il valore booleano risultante nella variabile boolean z. Scrivete l'istruzione:

z=Character.isWhitespace(c)

8. Scrivete un (frammento di) programma Java che operi come segue: legge 100 interi da tastiera e chiede all'utente di inserire due estremi i e j (che assumiamo essere compresi fra 0 e 99). Quindi, il programma stampa *i numeri inseriti il cui indice è compreso fra i e j* (è inteso che l'indice del primo intero inserito è 0, l'indice del secondo è 1, ..., l'indice dell'ultimo è 99).

   Ecco un esempio di esecuzione (le parti in grassetto sono state inserite dall'utente):

```
Numero 0:  12
Numero 1:  1288
Numero 2:  342
Numero 3:  342
Numero 4:  -23413
...
Numero 99:  23

Primo estremo:  1
Secondo estremo:  3

1288
342
342
```

```
int x[] = new int[ 100 ];
for ( int i = 0; i < 100; i++ )
 x[ i ] = in.readInt( "Numero " + i + ": " );
int a = in.readInt( "Primo estremo: " );
int b = in.readInt( "Secondo estremo: " );
for ( int i = a; i <= b; i++ )
 out.println( x[ i ] );
```

9. Scrivete un programma Java che legga una stringa (assumendo che sia non vuota) e stampi quante volte il primo carattere è ripetuto nella stringa. Per esempio, se la stringa fosse *carico* dovrebbe stampare 2 (perché il carattere c compare 2 volte).

```
String s;
int i, n, quante;
s = in.readLine();
n = s.length();
for ( i = quante = 0; i < n; i++ )
 if ( s.charAt( 0 ) == s.charAt( i ) )
  quante++;
out.println( quante );
```