

Informatica Generale

12 luglio 2005

Cognome Nome
Matricola

- Quando vi è richiesto di scrivere un programma, potete limitarvi al *corpo* del metodo `main`, assumendo se necessario che `in` e `out` siano due variabili di classe `ConsoleInputManager` e `ConsoleOutputManager` (rispettivamente), già dichiarate e inizializzate.
1. (a) Eseguire la sottrazione $37 - 54$ rappresentando i numeri in *complemento a 2* su 8 bit. Mostrare che il risultato rappresenta effettivamente -17 .
(b) Qual è la rappresentazione di -57 in *complemento a due* su 8 bit?
(c) Rappresentate il numero 44 in base 4.

(Svolgimento sul retro)

2. Per ciascuna delle seguenti espressione booleane, dite su quante righe della tabella di verità il valore dell'espressione è *vero* (\wedge = AND, \vee = OR, \oplus = XOR = OR ESCLUSIVO, \neg = NOT):

(a) $\neg x \wedge (x \vee (\neg x \vee \neg y))$:

(b) $x \vee (y \vee (x \vee y))$:

(c) $(\neg x \wedge y) \oplus (z \wedge \neg y)$:

(Usate il retro per le tabelle di verità)

3. Scrivete un (frammento di) programma Java che operi come segue:

- legga un numero intero, diciamo n , da tastiera;
- legga n stringhe da tastiera, memorizzandole in un array a ,
- stampi le sole stringhe successive alla prima introdotta che iniziano con la stessa lettera con cui la precedente terminata.

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

```
Quante stringhe: 8
Stringa 1: caro
Stringa 2: otto
Stringa 3: si
Stringa 4: può
Stringa 5: dire
Stringa 6: essenzialmente
Stringa 7: e
Stringa 8: sostanzialmente
```

```
otto
essenzialmente
e
```

(Svolgimento sul retro)

4. Scrivete un (frammento di) programma Java che operi come segue:

- legga un intero n da tastiera;
- legga n interi non negativi; ogni volta che l'utente inserisce un intero negativo, il programma deve segnalare un messaggio di errore, e chiedere all'utente di ripetere l'inserimento (finché il numero inserito non sia ≥ 0);
- al termine, stampi il numero dei valori inseriti che si discostano dalla media per più di 5.

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

```
Quanti valori: 5
Valore 0: 5
Valore 1: -4
Errore di inserimento; ripeti, prego
Valore 1: -3
Errore di inserimento: ripeti, prego
Valore 1: 15
Valore 2: 5
Valore 3: 1
Valore 4: 9

Risultato: 2
```

Commento: In questo esempio la media è $(5 + 15 + 5 + 1 + 9)/5 = 35/5 = 7$ e quindi i valori che si discostano dalla media per più di 5 sono 15 ($|15 - 7| = 8 > 5$) e 1 ($|1 - 7| = 6 > 5$).

Suggerimento: La classe `Math` contiene un metodo statico di nome `abs` che, preso come argomento un intero, ne restituisce il valore assoluto. La sua segnatura è:

```
public static int abs(int x)
```

(Svolgimento sul retro)

5. Assumete che x e y siano variabili `int`, e che il loro valore *prima di ciascuno dei seguenti assegnamenti* sia rispettivamente 3 e 5. Dite quale sarà il loro valore *dopo* gli assegnamenti indicati:

Assegnamento	x	y
$x=y+x+y$		
$y=++x$		
$x=x/y$		
$x=y+y++$		
$x*=y$		

6. Scrivete la classe `Treno` le cui istanze rappresentano dei collegamenti ferroviari; questa classe deve avere un costruttore pubblico

```
public Treno(String stazPart, String stazArr, int oraPart, int oraArr)
```

che costruisce un treno specificando stazione di partenza, di arrivo, orario di partenza e di arrivo: gli orari sono forniti mediante il numero di minuti da mezzanotte (ad esempio, se il treno parte alle 10:15 del mattino, l'orario di partenza sarà 615).

Dovete inoltre aggiungere i seguenti metodi:

- `int tempoPercorrenza()`: restituisce il tempo di percorrenza, in minuti: notate che se l'orario di arrivo precede l'orario di partenza significa che l'arrivo avviene il giorno dopo; ad esempio, un treno che parte alle 22:15 e arriva alle 9:15 ha un tempo di percorrenza di 11 ore;
- `boolean parteDiMattina()`: restituisce `true` se e solo se il treno parte di mattina (cioè, prima delle ore 12:00);
- `boolean impiegaMenoTempoDi(Treno t) throws TrenoException`: restituisce `true` se il treno che invoca il metodo impiega meno tempo del treno `t`, `false` altrimenti, *a patto che i due treni percorrano la stessa tratta*. Se i due treni precorrono tratte diverse (cioè se la stazione di partenza è diversa, o quella di arrivo è diversa, o entrambe le cose), il metodo deve sollevare un'eccezione.

(Svolgimento sul retro)