

Errata corrige del libro “Fondamenti di Informatica in Java”

Emilio Di Giacomo, Walter Didimo

Capitolo 1

	Errata	Corrige
Pag. 23, Figura 1.2	<p>The diagram shows a central 'Bus interno' (internal bus) connected to several components: a stack of registers (R1, R2, R3, ..., Rn), a Program Counter (PC), Instruction Register (IR), Program Status Word (PSW), and a Control Unit (Unità di controllo). Below the bus are the ALU, Memory Address Register (MAR), and Memory Data Register (MDR). In the 'Errata' version, the bus is connected to the ALU, MAR, and MDR. The labels below the bus indicate: 'al bus dati' (to the data bus) pointing to the MAR, 'al bus indirizzi' (to the address bus) pointing to the MDR, and 'al bus di controllo' (to the control bus) pointing to the Control Unit.</p>	
	<p>This diagram is identical to the one above but with corrected bus connections. The bus is now connected to the ALU, MAR, and MDR. The labels below the bus indicate: 'al bus indirizzi' (to the address bus) pointing to the MAR, 'al bus dati' (to the data bus) pointing to the MDR, and 'al bus di controllo' (to the control bus) pointing to the Control Unit.</p>	

Capitolo 2

	Errata	Corrige
Pag. 46, riga 7	$[-2^{k-1} - 1, 2^{k-1} - 1]$	$[-2^{k-1} + 1, 2^{k-1} - 1]$
Pag. 46, riga 12	$[-2^{k-1} - 1, 2^{k-1} - 1]$	$[-2^{k-1} + 1, 2^{k-1} - 1]$
Pag. 49, riga 21	Un calcolo con risultato minore di $1,00 \cdot 10^{-99}$ (o maggiore di $-1,00 \cdot 10^{-99}$)	Un calcolo con risultato positivo minore di $1,00 \cdot 10^{-99}$ (o negativo maggiore di $-1,00 \cdot 10^{-99}$)
Pag. 55, Esercizio 2.5	$10011\ 001 _2$	$10011001 _2$
Pag. 56, Esercizio 2.6	rappresentazione ottale	rappresentazione esadecimale
Pag. 56, Esercizio 2.6	$10111\ 0 _2$	$101110 _2$
Pag. 56, Esercizio 2.7	i seguenti numeri negativi	i seguenti numeri interi

Capitolo 5

	Errata	Corrige
Pag. 96, riga 17	vari directory	varie directory
Pag. 105, riga 36	classi e oggetto	classi e oggetti
Pag. 123, riga 6	la memoria che lui richiede	la memoria che esso richiede

Capitolo 6

	Errata	Corrige
Pag. 131, riga 25	<i>espressioni intera</i>	<i>espressione intera</i>
Pag. 139, riga 26	È ancora possibile comporre	È anche possibile comporre
Pag. 144, riga 11	la verità o falsità	la veridicità o la falsità
Pag. 144, ultima riga	due valori boolean	due valori boolean
Pag. 150, riga 31	<code>Integer.parseInt(100)</code>	<code>Integer.parseInt("100")</code>
Pag. 156, riga 23	Se il parametro <i>i</i>	Se il parametro <i>i</i>

Capitolo 7

	Errata	Corrige
Pag 179, Listato 7.3, linea 60	<code>double diffY = p2.coordX-p2.coordY</code>	<code>double diffY = p1.coordY-p2.coordY</code>

Capitolo 8

	Errata	Corrige
Pag. 209, ultime due righe	<pre>while (<condizione>) <istruzione>;</pre>	<pre>while (<condizione>) <istruzione></pre>
Pag. 210, righe 5-10	<pre>while (<condizione>){ <istruzione 1>; <istruzione 2>; ... <istruzione k>; }</pre>	<pre>while (<condizione>){ <istruzione 1> <istruzione 2> ... <istruzione k> }</pre>
Pag. 210, riga 33	il valore di <i>i</i> si mantiene	il valore di <i>i</i> si mantiene
Pag. 211, righe 25-27	<pre>do <istruzione>; while (<condizione>;</pre>	<pre>do <istruzione> while (<condizione>;</pre>
Pag. 212, righe 17-18	<pre>for (<inizializzazione>; <condizione>; <aggiornamento>) <istruzione>;</pre>	
	<pre>for (<inizializzazione>; <condizione>; <aggiornamento>) <istruzione></pre>	
Pag. 213, righe 4-5	<pre>for (; <condizione>; <aggiornamento>) <istruzione>;</pre>	<pre>for (; <condizione>; <aggiornamento>) <istruzione></pre>
Pag. 213, righe 7-8	<pre>for (<inizializzazione>; ; <aggiornamento>) <istruzione>;</pre>	
	<pre>for (<inizializzazione>; ; <aggiornamento>) <istruzione></pre>	
Pag. 213, righe 10-11	<pre>for (; ;) <istruzione>;</pre>	<pre>for (; ;) <istruzione></pre>
Pag. 214, righe 34-35	<pre>for (<inizializzazione>; <condizione>; <aggiornamento>) <istruzione>;</pre>	
	<pre>for (<inizializzazione>; <condizione>; <aggiornamento>) <istruzione></pre>	
Pag. 215, righe 1-4	<pre>while (<condizione>){ <istruzione>; <aggiornamento>; }</pre>	<pre>while (<condizione>){ <istruzione> <aggiornamento> }</pre>
Pag. 215, righe 9-10	<pre>while (<condizione>) <istruzione>;</pre>	<pre>while (<condizione>) <istruzione></pre>
Pag. 215, righe 12-16	<pre>if (<condizione>){ do <istruzione>; while (<condizione>); }</pre>	<pre>if (<condizione>){ do <istruzione> while (<condizione>); }</pre>
Pag. 215, riga 28	quando non è noto prevedere	non è possibile prevedere

	Errata	Corrige
Pag. 216, righe 28-33	<pre>for (int i = 0; i<=100; i++){ if (i<soglia) continue; else System.out.println(i); }</pre>	<pre>for (int i = 0; i<=100; i++){ if (i<soglia) continue; System.out.println(i); }</pre>

Capitolo 9

	Errata	Corrige
Pag. 236 (nota a pie' di pagine)	necessita un po' più di memoria	necessita di un po' più di memoria
Pag. 242, righe 6-8	<pre>int[] t = new int[s.length]; for (int i = 0; i<s.length; i++) t[i] = s[i];</pre>	
	<pre>String[] t = new String[s.length]; for (int i = 0; i<s.length; i++) t[i] = s[i];</pre>	
Pag. 243, riga 14	<pre>public massimo()</pre>	<pre>public int massimo()</pre>
Pag. 243, riga 16	<pre>public contiene(int k)</pre>	<pre>public boolean contiene(int k)</pre>
Pag. 246, riga 14	Il metodo massimo	Il metodo massimo
Pag. 248, riga 10	Il metodo contiene	Il metodo contiene
Pag. 248, righe 18-27	<pre>public boolean contiene(int k){ boolean esiste = false; int i = 0; while (!esiste && i<this.seq.length){ if (this.seq[i]==k) esiste = true; // elemento trovato i++; } return esiste; }</pre>	
	<pre>public boolean contiene(int k){ boolean esiste = false; int i = 0; while (!esiste && i<this.seq.length){ if (this.seq[i]==k) esiste = true; // elemento trovato i++; } return esiste; }</pre>	
Pag. 248, riga 31	che la variabile esiste	che la variabile <i>esiste</i>
Pag. 248, riga 35	Il metodo equals	Il metodo equals

	Errata	Corrige
Pag. 253, listato 9.6	<pre>public class Somma{ public static void main(String[] args){ if (args.length<2) System.out.println("Numero di parametri incorretto!"); else{ Integer n1 = Integer.valueOf(args[0]); Integer n2 = Integer.valueOf(args[1]); System.out.println(n1.intValue()+n2.intValue()); } } }</pre>	<pre>public class Somma{ public static void main(String[] args){ if (args.length<2) System.out.println("Numero di parametri incorretto!"); else{ Integer n1 = Integer.valueOf(args[0]); Integer n2 = Integer.valueOf(args[1]); System.out.println(n1.intValue()+n2.intValue()); } } }</pre>
Pag. 257, riga 19	a[3] = new double[4];	a[3] = new double[1];
Pag. 262, riga 13	la cui matrice rappresentata sia il prodotto	la cui matrice rappresentata è il prodotto
Pag. 263, riga 222	Il metodo somma	Il metodo somma
Pag. 263, riga 38	Il metodo prodotto	Il metodo prodotto
Pag. 264, riga 21	Il metodo toString	Il metodo toString

Capitolo 10

	Errata	Corrige
Pag. 283, riga 2	<code>new Magazzino();</code>	<code>new Magazzino(3);</code>
Pag. 302, Esercizio 10.1, righe 1-2	<code>CodiceFiscale, Nome, Cognome</code>	<code>codiceFiscale, nome, cognome</code>
Pag. 302, Esercizio 10.2, riga 7	<code>ProvaListaPersone</code>	<code>ProvaElencoPersone</code>
Pag. 303, Esercizio 10.5, riga 11	<code>uan</code>	<code>una</code>
Pag. 304, Esercizio 10.7, riga 11	<code>protected giorno</code>	<code>protected int giorno</code>
Pag. 304, Esercizio 10.7, riga 12	<code>protected mese</code>	<code>protected int mese</code>
Pag. 304, Esercizio 10.7, riga 13	<code>protected anno</code>	<code>protected int anno</code>

Capitolo 12

	Errata	Corrige
Pag 342, riga 21	identificato dalla condizione $i \leq 1$	identificato dalla condizione $n \leq 1$
Pag 350, riga 25	<code>r = m(y)</code>	<code>r = m_ric(y)</code>

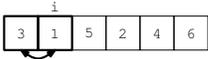
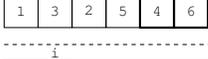
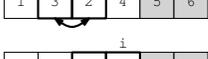
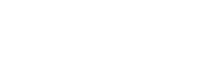
Capitolo 13

	Errata	Corrige
Pag. 358, listato 13.1	<pre>public static boolean tuttiPari(int[] a){ boolean tuttiPari = true; int i = 0; while (i<a.length){ if (a[i]%2!=0) tuttiPari = false; i++; } return tuttiPari; }</pre>	
	<pre>public static boolean tuttiPari(int[] a){ boolean tuttiPari = true; int i = 0; while (i<a.length && tuttiPari){ if (a[i]%2!=0) tuttiPari = false; i++; } return tuttiPari; }</pre>	
Pag 375, riga 32	sia $f_I(n)$ il numero di volte che viene eseguita l'istruzione I nell'ambito dell'esecuzione di M	sia $f_I(n)$ il costo dell'esecuzione dell'istruzione I nell'ambito dell'esecuzione di M
Pag 377, Algoritmo 1	i primi i elementi di a	i primi $i + 1$ elementi di a
Pag 378, riga 1	i elementi (per sommarli)	$i + 1$ elementi (per sommarli)

Capitolo 14

	Errata	Corrige
Pag. 387, righe 24–27	il numero di confronti nel caso peggiore è dato dal più piccolo intero h tale che $\frac{n}{2^h} \leq 1$, che può essere riscritta come $n \leq 2^h$ da cui, prendendo il logaritmo in base due di entrambi i membri, si ottiene $h \geq \log_2 n$. Visto che h è il più piccolo intero che soddisfa la disequazione precedente abbiamo $h = \lceil \log_2 n \rceil$.	il numero di confronti nel caso peggiore è dato dal più piccolo intero h tale che $\frac{n}{2^h} < 1$, che può essere riscritta come $n < 2^h$ da cui, prendendo il logaritmo in base due di entrambi i membri, si ottiene $h > \log_2 n$. Visto che h è il più piccolo intero che soddisfa la disequazione precedente abbiamo $h = \lfloor \log_2 n \rfloor + 1$.
Pag. 392, Listato 14.5, riga 12	<code>a[j] = a[j-1];</code>	<code>a[j] = a[j-1];</code> <code>j--;</code>

Pag. 397, Fig. 14.5

	Errata	Corrige
	 <p>daOrdinare=6 ultimoScambio=0 $a[i] < a[i-1]$</p>	
	 <p>daOrdinare=6 ultimoScambio=1 $a[i] > a[i-1]$</p>	
	 <p>daOrdinare=6 ultimoScambio=1 $a[i] < a[i-1]$</p>	
	 <p>daOrdinare=6 ultimoScambio=3 $a[i] < a[i-1]$</p>	
	 <p>daOrdinare=6 ultimoScambio=4 $a[i] > a[i-1]$</p>	
	<hr/>  <p>daOrdinare=4 ultimoScambio=0 $a[i] > a[i-1]$</p>	
	 <p>daOrdinare=4 ultimoScambio=0 $a[i] < a[i-1]$</p>	
	 <p>daOrdinare=4 ultimoScambio=2 $a[i] > a[i-1]$</p>	
	<hr/>  <p>daOrdinare=2 ultimoScambio=0 $a[i] > a[i-1]$</p>	
	<hr/>  <p>daOrdinare=0</p>	
	<hr/>  <p>daOrdinare=6 ultimoScambio=0 $a[i] < a[i-1]$</p>	
	 <p>daOrdinare=6 ultimoScambio=1 $a[i] > a[i-1]$</p>	
	 <p>daOrdinare=6 ultimoScambio=1 $a[i] < a[i-1]$</p>	
	 <p>daOrdinare=6 ultimoScambio=3 $a[i] < a[i-1]$</p>	
	 <p>daOrdinare=6 ultimoScambio=4 $a[i] > a[i-1]$</p>	
	<hr/>  <p>daOrdinare=4 ultimoScambio=0 $a[i] > a[i-1]$</p>	
	 <p>daOrdinare=4 ultimoScambio=0 $a[i] < a[i-1]$</p>	
	<hr/>  <p>daOrdinare=4 ultimoScambio=2 $a[i] > a[i-1]$</p>	
	<hr/> <p>daOrdinare=2 ultimoScambio=0 $a[i] > a[i-1]$</p>	
	<hr/> <p>daOrdinare=0</p>	

	Errata	Corrige
Pag. 398, Listato 14.8, linea 3	<code>int[] temp = new int[n];</code>	<code>int[] tmp = new int[n];</code>
Pag. 398, Listato 14.8, linea 8	<code>sx+1<dx</code>	<code>sx<dx</code>
Pag. 406, Listato 14.11, linea 7	<code>sx+1<dx</code>	<code>sx<dx</code>
Pag. 409, riga 20	$= T(n - i) + i \cdot c_2 \cdot n - \frac{(i-1)i}{2}$	$= T(n - i) + i \cdot c_2 \cdot n - \frac{(i-1)i}{2} c_2$
Pag. 409, riga 22	$T(n) = c_1 + c_2 \cdot n(n - 1) - \frac{(n-2)(n-1)}{2} = \frac{n^2}{2} + \frac{n}{2} - 1 = O(n^2)$	$T(n) = c_1 + c_2 \cdot n(n - 1) - \frac{(n-2)(n-1)}{2} c_2 = c_2 \frac{n^2}{2} + c_2 \frac{n}{2} + (c_1 - c_2) = O(n^2)$

Capitolo 15

	Errata	Corrige
Pag. 421, List. 15.3, riga 4	private final int DIM_INIZIALE = 10	private static final int DIM_INIZIALE = 10
Pag. 433, linea 1	ed vuota()	e vuota()
Pag. 433, linea 5	soltanto i riferimento	soltanto il riferimento
Pag. 433, linea 7	index	indice
Pag. 434, linea 9	ad lunghezza-2	a lunghezza-2
Pag. 442, Fig. 15.11 (b)		
Pag. 442, Fig. 15.11 (c)		
Pag. 442, Fig. 15.11 (d)		
Pag. 444, righe 6 e 9	rimuoviUltimo(...)	rimuoviUltimo()

Capitolo 16

	Errata	Corrige
Pag. 505, Esercizio 16.3, riga 10	una classe <code>Ora</code> che implementa <code>dotata</code> di tre campi	una classe <code>Ora</code> dotata di tre campi

Capitolo 17

	Errata	Corrige
Pag. 509, righe 4-17	<pre>public String toString(){ String s = ""+valore; switch(seme){ case COPPE: s += " di coppe"; case SPADE: s += " di spade"; case BASTONI: s += " di bastoni"; case DENARI: s += " di denari"; } return s; }</pre>	
	<pre>public String toString(){ String s = ""+valore; switch(seme){ case COPPE:{ s += " di coppe"; break; } case SPADE:{ s += " di spade"; break; } case BASTONI:{ s += " di bastoni"; break; } case DENARI:{ s += " di denari"; break; } } return s; }</pre>	
Pag. 511, riga 24	l'operatore	l'operatore

Capitolo 18

	Errata	Corrige
Pag. 528, riga 2	l'array a nell'array b	l'array a nell'array b
Pag. 534, riga 11	Le strategia più adeguate	Le strategie più adeguate
Pag. 536, riga 6 e 9	<code>str.split("");</code>	<code>str.split(" ");</code>

Capitolo 19

	Errata	Corrige
Pag. 552, Listato 19.4, righe 14-17	<pre>while(i<args.length){ ifout.writeInt(Integer.valueOf(args[i])); i++; }</pre>	
	<pre>while(i<args.length){ ifout.writeInt(Integer.valueOf(args[i])); i++; }</pre>	