

Laboratorio di Programmazione
Corso di laurea triennale in Informatica Musicale
Università degli Studi di Milano, A.A. 2017-2018

Nicola Basilico, Andrea Lanzi

Appello del 22 Gennaio 2018

- L'esame ha una durata di 3 ore e 30 minuti.
- È possibile consultare il libro di testo, appunti e l'API Java.
- È possibile utilizzare qualsiasi ambiente di sviluppo installato sulla macchina.
- È proibito l'accesso ad Internet con qualsiasi mezzo.
- Ogni esercizio consegnato non deve generare errori di compilazione.
- Per ottenere la sufficienza è necessario consegnare almeno due esercizi corretti.

1 Scambio di elementi in un array

nome del file sorgente: *Scambio.java*

Si scriva un programma che:

- acquisisca due array $A1$ e $A2$ ciascuno contenente dieci numeri interi;
- popoli un terzo array $A3$ facendo il massimo elemento per elemento dei primi due e cioè per $i = 1, 2, \dots, 10$ $A3[i]$ sarà pari al massimo tra $A1[i]$ e $A2[i]$;
- scambi, in $A3$, ogni elemento in posizione pari con l'elemento successivo.

1.1 Esempio

Acquisiti questi due array:

```
A1 = 7 3 8 4 12 3 9 1 1 1
A2 = 1 1 5 12 12 4 7 9 0 18
```

Il programma calcola:

```
A3 = 7 3 8 12 12 4 9 9 1 18
```

E infine:

```
A3 = 3 7 12 8 4 12 9 9 18 1
```

2 Stampa vocali e consonanti

nome del file sorgente: *Sottostringhe.java*

Si scriva un programma che prenda in input due stringhe S e P di lunghezza pari a 6 caratteri e crei una prima sottostringa $S1$ contenente solo le vocali di S ed una sottostringa $S2$ con le consonanti di P . Il programma concateni poi le due sottostringhe ottenute in una stringa finale e la stampi in maiuscolo. Si utilizzi la funzione `charAt()` messa a disposizione dal linguaggio Java.

2.1 Esempio

Acquisite queste due stringhe:

```
S = canile
P = casata
```

il programma calcola:

```
S1 = aie
S2 = cst
```

Infine il programma stampa la stringa:

```
AIECST
```

3 Matrice

nome del file sorgente: Matrice.java

Si scriva un programma che acquisisca una matrice di interi di dimensione 3×3 . Il programma poi stampi a video un menu che permetta di scegliere una tra queste operazioni:

1. stampare la diagonale della matrice
2. stampare l'antidiagonale della matrice
3. stampare il massimo elemento della matrice
4. stampare il minimo elemento della matrice
5. stampare la matrice trasposta (ottenuta scambiando righe con colonne)

3.1 Esempio

Acquisita questa matrice:

```
1 2 3
4 5 6
7 8 9
```

diagonale:

```
1 5 9
```

antidiagonale:

```
3 5 7
```

massimo:

```
9
```

minimo:

```
1
```

trasposta:

```
1 4 7
2 5 8
3 6 9
```

4 Controllo Protocollo

nome del file sorgente: ControlloProtocollo.java

Il programma *ControlloProtocollo* legge un pacchetto di rete in formato stringa inserito dall'utente ed effettua una sequenza di test per verificarne la validità rispetto al protocollo. Come risultato stampa un rapporto che indica, per ogni test effettuato, se il protocollo è rispettato o meno. Vengono effettuati quattro diversi test, ciascuno dei quali è definito come la verifica di una condizione sul pacchetto inserito. I test verificano se la stringa che rappresenta il pacchetto:

1. è composto da almeno 15 caratteri (TEST 1);
2. contiene almeno due maiuscole consecutive e almeno una cifra decimale (0,1, ..., 9) (TEST 2);
3. contiene consonanti in tutte le posizioni pari (TEST 3);
4. i caratteri in posizioni pari formano una stringa palindroma (case sensitive) (TEST 4).

Si fornisca un'implementazione del programma che, una volta letto il pacchetto, stampi il rapporto e termini.

4.1 Esempio di esecuzione

L'utente inserisce questo pacchetto:

```
B6Gza4t6XHtHauGiB
```

Il programma stampa questo report:

```
TEST 1: ok
TEST 2: ok
TEST 3: fallito
TEST 4: ok
```

L'utente inserisce questo pacchetto:

```
1cGz
```

Il programma stampa questo report:

```
TEST 1: fallito
TEST 2: fallito
TEST 3: fallito
TEST 4: fallito
```

4.2 Info utili

Estratto della tabella ASCII.

CODICE	CARATTERE	DESCRIZIONE
32	(space)	(space)
33	!	(exclamation mark)
34	"	(Quotation mark)
35	#	(Number sign)
36	\$	(Dollar sign)
37	%	(Percent sign)
38	&	(Ampersand)
39	'	(Apostrophe)
40	((round brackets or parentheses)
41)	(round brackets or parentheses)
42	*	(Asterisk)
43	+	(Plus sign)
44	,	(Comma)
45	-	(Hyphen)
46	.	(Full stop , dot)
47	/	(Slash)
48	0	(number zero)
49	1	(number one)
50	2	(number two)
51	3	(number three)
52	4	(number four)
53	5	(number five)
54	6	(number six)
55	7	(number seven)
56	8	(number eight)
57	9	(number nine)
58	:	(Colon)
59	;	(Semicolon)
60	<	(Less-than sign)
61	=	(Equals sign)
62	>	(Greater-than sign)
63	?	(Question mark)
64	@	(At sign)
65	A	(Capital A)
66	B	(Capital B)
67	C	(Capital C)
68	D	(Capital D)
69	E	(Capital E)
70	F	(Capital F)
71	G	(Capital G)

CODICE	CARATTERE	DESCRIZIONE
72	H	(Capital H)
73	I	(Capital I)
74	J	(Capital J)
75	K	(Capital K)
76	L	(Capital L)
77	M	(Capital M)
78	N	(Capital N)
79	O	(Capital O)
80	P	(Capital P)
81	Q	(Capital Q)
82	R	(Capital R)
83	S	(Capital S)
84	T	(Capital T)
85	U	(Capital U)
86	V	(Capital V)
87	W	(Capital W)
88	X	(Capital X)
89	Y	(Capital Y)
90	Z	(Capital Z)
91	[(square brackets)
92	\	(Backslash)
93]	(square brackets)
94	^	(Caret or circumflex accent)
95	_	(underscore)
96	‘	(Grave accent)
97	a	(Lowercase a)
98	b	(Lowercase b)
99	c	(Lowercase c)
100	d	(Lowercase d)
101	e	(Lowercase e)
102	f	(Lowercase f)
103	g	(Lowercase g)
104	h	(Lowercase h)
105	i	(Lowercase i)
106	j	(Lowercase j)
107	k	(Lowercase k)
108	l	(Lowercase l)
109	m	(Lowercase m)
110	n	(Lowercase n)
111	o	(Lowercase o)
112	p	(Lowercase p)
113	q	(Lowercase q)
114	r	(Lowercase r)
115	s	(Lowercase s)

CODICE	CARATTERE	DESCRIZIONE
116	t	(Lowercase t)
117	u	(Lowercase u)
118	v	(Lowercase v)
119	w	(Lowercase w)
120	x	(Lowercase x)
121	y	(Lowercase y)
122	z	(Lowercase z)
123	{	(curly brackets or braces)
124		(vertical-bar)
125	}	(curly brackets or braces)
126	~	(Tilde)
