

1 Massimo pari e dispari di una sequenza

Si scriva un programma (*sequenza_max_pari_dispari.java*) che legga in input una sequenza di interi terminata da 0. Ad esempio la sequenza:

```
-15 -7 5 -12 3 0
```

Ogni volta che un nuovo intero diverso da zero viene letto, il programma stampa un messaggio che riporta il massimo numero pari e il massimo numero dispari letti fino a quel momento. Se nessun numero pari o nessun numero dispari è stato letto viene stampato “(nessuno)”.

1.1 Esempio di esecuzione

Il programma stampa:

```
Inserire il primo intero:
```

l'utente inserisce:

```
-15
```

il programma stampa:

```
Il massimo pari è (nessuno).
```

```
Il massimo dispari è -15.
```

```
Inserire il prossimo numero:
```

l'utente inserisce:

```
-7
```

il programma stampa:

```
Il massimo pari è (nessuno).
```

```
Il massimo dispari è -7.
```

```
Inserire il prossimo numero:
```

l'utente inserisce:

```
5
```

il programma stampa:

```
Il massimo pari è (nessuno).
```

```
Il massimo dispari è 5.
```

```
Inserire il prossimo numero:
```

l'utente inserisce:

-12

il programma stampa:

Il massimo pari è -12.

Il massimo dispari è 5.

Inserire il prossimo numero:

l'utente inserisce:

3

il programma stampa:

Il massimo pari è -12.

Il massimo dispari è 5.

Inserire il prossimo numero:

l'utente inserisce:

0

il programma termina.

2 La congettura di Collatz (collatz.java)

2.1 Prima parte

L'algoritmo di Collatz genera, dato un intero positivo n , una serie numerica. La sua definizione è la seguente:

1. aggiungi n alla serie;
2. se n è pari a 1, termina;
3. se n è pari, aggiungi $q=n/2$ alla serie;
4. se n è dispari, aggiungi $q=3n+1$ alla serie;
5. assegna $n=q$ e riparti dal punto 2.

Si scriva un programma che, dato n , stampi a video la serie ottenuta dall'algoritmo descritto sopra per un massimo di 20 elementi.

2.1.1 Esempi di test

Dato in input:

5

il programma stampa:

Serie calcolata: 5 ,16, 8, 4, 2, 1;

Dato in input:

1

il programma stampa:

Serie calcolata: 1;

Dato in input:

27

il programma stampa:

Serie calcolata:

27 ,82, 41, 124, 62, 31, 94, 47, 142, 71,

214, 107, 322, 161, 484, 242, 121, 364, 182, 91

(serie troncata, raggiunto numero massimo di elementi)

2.2 Seconda parte

Si estenda poi il programma in modo che presenti un menu all'utente con il quale scegliere se generare una serie dato n (prima parte dell'esercizio) o che, data una sequenza di interi, verifichi se è stata generata dall'algoritmo. Si assuma che la sequenza data in input termini con un 1 o sia di al più 20 elementi.

2.2.1 Esempi di test

Dato in input:

1

il programma stampa:

La serie è ammissibile.

Dato in input:

27 82 41 124 62 31 95 1

il programma stampa:

La serie NON è ammissibile.

E' stato inserito 95, dove sarebbe dovuto comparire 94.

Dato in input:

19 58 29 88 44 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2

il programma stampa:

La serie è ammissibile.

3 Atoi Function (atoiFun.java)

Implementare la funzione atoi ascii to integer in java usando la funzione charAt(). Inserire 2 stringhe rappresentanti 2 numeri a 3 cifre ed eseguire la somma tra i 2 numeri stampando il risultato.

4 Funzioni strpbrk strstr (strFunctions.java)

Implementare la funzione strpbrk che prende in input due stringhe: str1 e str2 e restituisce il primo carattere nella stringa str1 che corrisponde ad uno qualsiasi dei caratteri contenuti nella stringa str2, usare la funzione charAt() di java.

Esempio: str1 = "abcde2fghi3jk4l"; str2 = "34" ; stampa "3"

Implementare la funzione strstr che prende in input due stringhe str1 e str2 e trova la prima occorrenza della stringa str2 nella stringa str1 e restituisce la prima posizione della stringa trovata.

Esempio: str1 = "Pointless"; str2 = "less" ; stampa "5" ;

5 Rettangolo di asterischi

Si scriva un programma (*rettangolo_asterischi.java*) che dati due interi b e h disegni un rettangolo di asterischi di base b e altezza h .

5.1 Esempio di esecuzione

L'utente inserisce la base

5

e l'altezza

10

il programma stampa

```
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****
```


6 Istogramma di asterischi

Il programma *istogramma_asterischi.java* stampa un istogramma orizzontale a partire dai dati inseriti dall'utente. All'inizio viene chiesto all'utente il numero di barre dell'istogramma $N \geq 0$. Successivamente, il programma chiede il valore (strettamente positivo) per ciascuna delle N barre e stampa un istogramma usando la stringa “[*]” per rappresentare le unità.

6.1 Esempio di esecuzione

All'avvio il programma stampa:

Inserire numero di barre:

l'utente inserisce:

5

dopodichè questa sequenza di dati è richiesta e inserita:

Inserire valore (positivo) della barra 1: 12

Inserire valore (positivo) della barra 2: 4

Inserire valore (positivo) della barra 3: 5

Inserire valore (positivo) della barra 4: 2

Inserire valore (positivo) della barra 5: 10

il programma quindi stampa questo istogramma orizzontale:

```
-----  
| [*] [*] [*] [*] [*] [*] [*] [*] [*] [*] [*]  
| [*] [*] [*] [*]  
| [*] [*] [*] [*] [*]  
| [*] [*]  
| [*] [*] [*] [*] [*] [*] [*] [*] [*] [*]  
-----
```

7 Gioco delle ultime due lettere (dueLettere.java)

Nel gioco delle ultime due lettere ci sono n giocatori che a turno, ciclicamente, scelgono una parola. Ogni giocatore, per restare nel gioco, deve scegliere una parola valida, cioè che:

1. inizi con le ultime due lettere della parola valida più recente;
2. sia diversa dalla parola valida più recente.

All'inizio del gioco per il giocatore 1 qualsiasi parola è valida. Se la parola scelta da un giocatore non è valida, tale giocatore è escluso dal gioco. Il gioco termina quando rimane un solo giocatore.

Si scriva un programma che simuli questo gioco per 4 giocatori chiedendo ciclicamente ai giocatori non ancora eliminati di scrivere la parola scelta e che alla fine stampi il numero del giocatore vincente.

7.1 Esempio di test

Data questa sequenza di parole in input:

```
mamma  
matrimonio  
iodio  
babbeo  
ionizzante  
testare  
riscontro  
remo  
moschettiere  
restituzione  
nospola  
errore
```

Il programma stamperà

```
IL GIOCATORE 2 VINCE
```

dopo aver eliminato, in ordine, i giocatori 4, 3 e 1