

1 Moltiplicazione senza moltiplicazione (moltiplicazione.java)

Si scriva un programma che dati due interi a e b stampi il risultato di $a \times b$ senza fare moltiplicazioni.

Suggerimento: si usi il metodo:

`Math.abs(i)`

per ottenere il valore assoluto di un intero i .

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 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.

3.1 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

4 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.

5 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" ;

6 Compositore di stringhe (`compositore.java`)

Il compositore di stringhe permette di generare stringhe a partire da codici numerici. Sia i un intero diverso da zero e sia $S(i)$ una stringa associata all'intero i . $S(i)$ è definita in questo modo: se $0 < i < 6$ allora $S(i)$ è una stringa inizialmente scelta dall'utente, altrimenti $S(i)$ è la stringa "[]".

Sia data una sequenza di interi di lunghezza arbitraria e terminata da 0

`x y z h k ... 0`

il compositore genera questa stringa

`<sep>S(x)<sep>S(y)<sep>S(z)<sep>S(h)<sep>S(k) ... <sep>`

dove è una stringa speciale scelta dall'utente detta "separator" .

Si scriva un programma che chieda all'utente le 5 stringhe da associare ai codici 1, 2, 3, 4 e 5, la stringa separator e una sequenza di interi e che stampi la stringa ottenuta secondo il meccanismo descritto sopra.

6.1 Esempi di test

Assegnando ai codici 1, 2, 3, 4 e 5 queste stringhe

`aa bb cc dd ee`

e assegnando al separator questa stringa

*

si ha:

sequenza di interi:

1 2 3 4 5 0

stringa generata

*aa*bb*cc*dd*ee*

sequenza di interi:

-20 4 12 3 6 3 -9 8 4 78 1 55 0

stringa generata

*[]*dd*[]*cc*[]*cc*[]*[]*dd*[]*aa*[]*

sequenza di interi:

9 9 9 9 9 9 0

stringa generata

[][]*[]*[]*[]*[]*[]*[]*

sequenza di interi:

0

nessuna stringa generata