

Ingegneria del Software

Esercitazione 4

Stack

Implementare la classe Stack con i Generics

Stack (II)

Aggiungere un Iteratore alla classe Stack

Iteratore di Fibonacci

$$F_1 = 1,$$

$$F_2 = 1,$$

$$F_n = F_{n-1} + F_{n-2} \text{ (per ogni } n > 2)$$

Polinomi

- Scrivere un iteratore che restituisca i coefficienti del polinomio, ordinati per grado
- **Esempio:**
- per $3 + 2x$ deve restituire la sequenza
3
2
NoSuchElementException

Functional

Functional ArrayList

Estendere ArrayList con i metodi

- map
- filter
- reduce

Takewhile

Estendere ArrayList con il metodo takewhile che prende un predicato e ritorna una nuovo ArrayList contente tutti gli elementi fino al primo elemento che viola il predicato.

```
var a = MyArrayList<String>();  
a.add("abc");  
a.add("d");  
a.add("efg");  
a.takewhile((i)->i.length() > 1); // ["abc"]  
a.takewhile((i)->i.length() > 0); // ["abc", "d", "efg"]  
a.takewhile((i)->i.length() > 5); // []
```


makeGreet

Creare un generatore di saluti il cui funzionamento ricalchi l'esempio seguente.

```
var a = Greet.make("Hello");  
a.accept("Giovanni"); // prints Hello Giovanni  
a.accept("Andrea"); // prints Hello Andrea  
var b = Greet.make("Ciao");  
b.accept("Silvia"); // prints Ciao Silvia
```

Multithreading

Conto corrente

Tre persone hanno un fondo comune. Una persona, il produttore, ha il compito di depositare i soldi per tutti alla fine del mese ma non può spenderli mentre le altre due, i consumatori, posso prelevare. Il conto non può andare in rosso.

Si implementino in Java 3 processi che simulino il fondo comune e gli accessi. Il produttore deposita max 200€ ogni 5 secondi, mentre il primo consumatore può prelevare max 300€ ogni secondo, mentre l'altro max 200€ ogni tre secondi.