

## Seminar 4

### Interfete functionale (Functional Interface)

1. Declarati o interfata functionala Arie. Definiti functii lambda pentru calculul ariilor unor figure geometrice. Implementati metoda definita de urmatoarea signatura:

```
public static <E> void printArie(List<E> l, Arie<E> f) {}
```

### Built-in functional interfaces

#### Predicates

2. Scrieti o metoda generica care afiseaza entitatile dintr-o lista, care satisfac un anumit criteriu, specificat ca parametru de tip Predicate.

#### Functions **apply**

3. Scrieti o declarație corespunzatoare pt:
  - a) `converterLambda=x->Integer.valueOf(x);`
  - b) `converterMethodReference Integer::valueOf;`

#### Suppliers **get**

4. Aceeasi cerința ca la punctul 3.

#### Comparators

5. Diferite exemple pentru sortari

### III. Stream

6. Scrieti o metoda generica care filtreaza o lista de obiecte dupa un anumit criteriu, specificat ca si un Predicate in lista de parametrii ai metodei

```
public <E> List<E> filterGeneric(List<E> lista, Predicate<E> p)
```
7. 2. Scrieti o metoda generica care filtreaza si sorteaza o lista obiecte de tip generic
8. 

```
public <E> List<E> filterGeneric(List<E> lista, Predicate<E> p,  
Comparator<E>)
```