Tutorial problems – Wed 30 Oct and Fri 1 Nov 2019

5min Ask whether there are any questions. Collect these by writing them on the board and decide with the whole group which of these to address in the 5 minutes available for question answering. The answering of the question may also be put to the end of the tutorial.

10min Write a method
`public static void arrayApply(int[] a, Function<Integer,Integer> f)` that applies the function `f` to each element in a. Possible answer:

```
public static void arrayApply(int[] a, Function<Integer,Integer> f) {
    for (int i = 0; i < a.length; i++) {
        a[i] = f.apply(a[i]);
    }
}
```

In main method you can then call e.g.:

```
int[] a = {1,5,7,4,5};
arrayApply(a, x -> {return x * x;});
arrayApply(a, x -> {return x + 1;});
```

10min Write a method

```
public static ArrayList<BankAccount> filter(
        ArrayList<BankAccount> allAccounts,
        Function<BankAccount,Boolean> f)
```

which allows to filter all accounts according to the function `f`, e.g., all ones with more than 100000 in the account, or all in debt.

5min Students to work in small groups (of 2 or 3) to answer:
"Explain the idea of Interfaces."

Expected answer: Contract for implementation. Interfaces contain only method headers but no implementation of the methods. Any implementing class has to provide full methods for the method headers. Can be used as types.

30min The students should again work in small group of 2 or 3 to address the following problem with respect to Interfaces (also involving arrays, loops, ...)

- First, implement a class `BankAccount` with only one field variable `private double balance` and the corresponding getter `public double getBalance()`.
- Second, implement a class `Country` with only one field variable `private double area` and the corresponding getter `public double getArea()`.

- Third, implement a class **`Statistics`** which contains a method **`public static double average(BankAccount[] bankAccounts)`** and a method **`public static double average(Country[] countries)`**. [The class would typically contain other methods related to median value, standard deviation and so forth, but they are of no concern here.]
- Introduce an Interface

```
public interface Measurable {
   public double getMeasure();
}
```

  Change the code in the three classes so that the classes **`BankAccount`** and **`Country`** implement **`Measurable`** and that the class **`Statistics`** avoids the redundancy of two methods which essentially compute the same.

© 2019, Alexandros Evangelidis, Manfred Kerber, Computer Science, University of Birmingham