Exercises: Week 6

Introductory Programming 2020

Testing

Exercise 0

Create or open a Gradle project ((Re)watch Troels creating a new project here).

Exercise 1

Write a simple Hello World program that returns a `String`. Then write a test case to check whether it returns the correct string.

Exercise 2

Write a program that has a method '`inverse`' which gets an `array` of `int` as input, and calculates the inverse of the array and returns it (Hint: a loop might be useful here). Then write a set of test cases to check if the inverse method works correctly.

For example:
`inverse([1,4,5,8,0,2,3,1,6])` should return the array `[6,1,3,2,0,8,5,4,1]`

Exercise 3

Write a method that concatenates two `Strings` and use a set of test cases to check the correctness of the implemented method.

Exercise 4

Implement a method with signature `int f(int x)`. If $x\%2 = 1$ then the method returns $x + x$, otherwise the method returns $x \cdot x$. Write a set of test cases that check the correctness of your implementation.

Exercise 5

Consider the following method that, given an `int` num, returns how many times we can divide it by 2 to get down to 1.

```
int divNum(int num) {
  int count = 0;
  while (num >= 1) {
    num = num / 2;
    count++;
  }
  return count;
}
```

Write a set of test cases that checks the correctness of the method.

Exercise 6

Consider the following program that calculates the reverse of a number:

```
public int reverseNum(int num){
    int reverse = 0;
    while(number != 0){
        reverse = (reverse*10)+(num%10);
        num = num/10;
    }
    return reverse;
}
```

Write a set of test cases to check the correctness of the implementation.

Exercise 7

Write a method that finds the minimum number in an `array` of `int` with size `Max`. Write a set of test cases to check that the method is correct.

Exercise 8 (Challenge!)

Consider the following method that sorts an array:

```
1  public static void bubbleSort(int[] a) {
2      boolean sorted = false;
3      int temp;
4      while(!sorted) {
5          sorted = false;
6          for (int i = 0; i < a.length; i++) {
7              if (a[i] > a[i+1]) {
8                  temp = a[i];
9                  a[i] = a[i+1];
10                 a[i+1] = temp;
11                 sorted = true;
12             }
13         }
14     }
15 }
```

Check if the sorting method is working correctly by writing a set of test cases. If you find bugs then correct the program such that it behaves correctly.

1 Inheritance and Polymorhism (Chapter 10)

Exercise 10.3

Draw an inheritance hierarchy for the people in your place of study or work. For example, if you are a university student, then your university probably has students (first-year students, second-year students, ...), professors, tutors, office personnel, etc.

Exercise 10.8

Open the network-v2 project. Add a class for event posts to the project. Create some event-post objects and test that all methods work as expected.

Exercise 10.9

Order these items into an inheritance hierarchy: apple, ice cream, bread, fruit, food item, cereal, orange, dessert, chocolate mousse, baguette.

Exercise 10.12

Assume that we have four classes: `Person`, `Teacher`, `Student`, and `PhDStudent`. `Teacher` and `Student` are both subclasses of `Person`. `PhDStudent` is a subclass of `Student`.

a) Which of the following assignments are legal, and why or why not?

- `Person p1 = new Student();`
- `Person p2 = new PhDStudent();`
- `PhDStudent phd1 = new Student();`
- `Teacher t1 = new Person();`
- `Student s1 = new PhDStudent();`

b) Suppose that we have the following legal declarations and assignments:

- `Person p1 = new Person();`
- `Person p2 = new Person();`
- `PhDStudent phd1 = new PhDStudent();`
- `Teacher t1 = new Teacher();`
- `Student s1 = new Student();`

Based on those just mentioned, which of the following assignments are legal, and why or why not?

- `s1 = p1`
- `s1 = p2`
- `p1 = s1`
- `t1 = s1`
- `s1 = phd1`
- `phd1 = s1`

Exercise 10.14

What has to change in the `NewsFeed` class when another `Post` subclass (for example, a class `EventPost`) is added? Why?

Exercise 10.15

Use the documentation of the Java standard class libraries to find out about the inheritance hierarchy of the collection classes. Draw a diagram showing the hierarchy.

Challenge

1. Open the 'University' folder provided on LearnIT. In it, the classes mentioned in Exercise 10.12 (above) have been created. Modify the classes such that they fulfill the inheritance requirements described in Exercise 10.12.

2. The constructor for class `Person` is already made but the rest still needs theirs. Create the constructor for each of the remaining classes. (Hint: Remember that the constructor of a subclass must always invoke the constructor of its superclass as its first statement.)

3. Create a method, `introduceYourself()` in class `Student`. It *does not* take any parameters. When called, the method should print out the message:

   ```
   "Hi!  My name is [name], I am [age] years old and I am currently
   following the [studyprogramme] programme at [university]!  Yay!"
   ```

   Where the values in brackets ([]) should change from student to student. (Hint: you do not need to add anything outside the method body to make this work).