Data Structures (CS 102), Professor Yap – Fall 2019

HOMEWORK

September 17, 2019

Homework 2: Due on Tuesday Sep 24, 2019

- Carefully read the Homework Policy in the class wiki. Pay attention to the rules for submitting programs in a zip file.

PART A: WRITTEN ASSIGNMENT

Question 1. (12 Points) Recall the class `Rectangle` and its subclass `LocatedRect` in homework 1. Consider the following code fragment:

```
Rectangle rr = new LocatedRect(1,2,3,4);
System.out.printf(rr.top());
System.out.printf(rr.toString());
```

Each line contains an “apparent error”. You must say what is the “apparent error”; also say whether Java considers it as an actual error.

Question 2. (4 Points) Class or Instance?
We have a valid Java program containing this statement:

```
if (X.Y > 0) System.out.println(‘‘Y is positive’’);
```

Is `X` the name of a class or the name of an instance?
HINT: is `Y` static or dynamic?

Question 3. (12 Points) Write the method `convert(int)` which converts an integer in the range 1-365 (representing a day in a year) into an array of two integers, representing month and day of the month. For instance, `convert(3)` should return the array of `[1,3]`, (i.e., 3rd of January); `convert(365)` should return the array of `[12,31]`, (i.e., 31st of December). Note: we are assuming a non-leap year (also called a ”common year”). You must use this method in Part B below.

Question 4. (4 Points) Odd One Out
All but one of the following words are roughly synonyms. Pick the one that does not belong in the list:

null, reference, link, address, pointer

Question 5. (8 Points)
For simplicity, we will not explicitly write packages in this course. So all the classes that you write are belong to the so-called “default package” (see p. 44 of Text). There are four access modifiers: public, protected, private and default.
(a) How is the the default access modifier different from the other three?
(b) Can all these modifiers be applied to any member? Any method? Any class?

PART B: PROGRAMMING ASSIGNMENT (40 Points)

B.1 The **Birthday Paradox** says that in any group of more than 23 people, the chance of two people sharing a birthday is more than 50%. We will test this idea empirically. This is **Project P-1.29** in Chapter 1 (p. 57) of our Text. You must use Random Numbers, naturally.

You must write two classes: `Bday` and `Paradox`. The class **Bday** represents each birthday as a pair (month, day) of integers. E.g. Washington's Birthday is $(2, 22)$ corresponding to February 22. It has one constructor `Bday(Random rg)` which is used to generate a random number in the range 1-365, then converted into a (month,day) pair. You must use the method you wrote in Part A. You must also write a `Bday.toString()` method that converts $(2, 22)$ to `"Feb 22"`.

B.2 The second class `Paradox` has a `main` method and will use the `Bday` class to test the Birthday paradox. The `main` method takes four command line arguments:

- `int ss` is the "seed" (with default value `ss=111`)
- `int nn` is the "size" (with default `nn=24`)
- `int mm` is the "(testing) mode" (with default `mm=0`)
- `int pp` is the "number of experiments" (with default `pp=100`)

If the mode is `mm=0`, then you generate `nn` many random birthdays and report success if any two birthdays are the same. If they are all different, print the list of birthdays. If the mode is `mm=1`, you will run the same test as `mm=0` for `pp` times, and simply report the number of successes.

B.3 The `Paradox` class has a static member:

```
static Random rg;
```

which we initialize using `ss` as the seed. However if `ss=0`, you must initialize `rg` without any seed (read up what this means). DO NOT INITIALIZE `rg` MORE THAN ONCE.

B.4 We have provides a zip file `hw2_Yap.zip` that contains a `Makefile` and a `src` folder where the skeletons of these two classes have been provided. Please fill them in with your own code. We can test your code using this Makefile, and you can also modify (as long as you do not modify our standard targets – there are 7 of them). We provided a sample output file `TestAllResult` containing output from our 7 targets. **You must aim to duplicate our outputs on these 7 targets!**