Assignment 1
Verifiable Procedural Program

Read Section 1 to understand the requirements, Section 2 to understand the programming tasks that you need to carry out and Section 3 to know the assessment detail and submission requirements.

1. Description

In this assignment, you will modify the design and implementation of the program CoffeeTinGame so that it behaves more like a proper procedural program. Your starting point is the high-level pseudocode of the program, which is repeated in Listing 1 below for your reference.

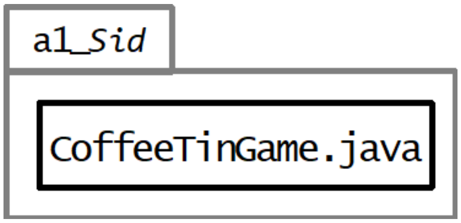

Figure 1: Program structure.

Listing 1: A high-level pseudocode of the coffee tin game

```
1.  while at least two beans in tin do
2.     take out any two beans
3.     if they are the same colour
4.        throw them both away
5.        put a blue bean back in (may be taken from an extra bag of beans)
6.     else
7.        throw away the blue one
8.        put the green one back
```

2. Task requirements

Complete the following tasks:

1. Create a package named a1_sid as shown in Figure 1, where sid is your student id. For example, if your student id is 123456789 then the package name is a1_123456789.

   You will need to use this package to store all the Java class(es) that you create for the program.

   IMPORTANT: Failure to name the package as described above will result in an invalid program.

2. Create in the above package a new Java program named CoffeeTinGame. This program must contains the procedures main and tinGame. However, you must perform the following enhancements so that the program behaves more like a proper game (the line numbers mentioned below refer to those in Listing 1):

   (a) Create a static constant named BeansBag that represents the extra bag of available beans. For testing purposes, BeansBag needs to be an array whose length is at least 30 which contains blue beans (character ‘B’), green beans (character ‘G’) and empty spaces (character ‘-‘), each type should accounts for roughly one third.

   Hint: Initialise BeansBag with a static initializer.

   (b) Change these 2 existing procedures to public: tinGame, takeOne.

   (c) Create 3 new public procedures (see details below): randInt, getBean, updateTin.

(d) Procedure `randInt`: accepts as input a positive integer *n* and returns as output an integer number randomly selected from the range [0, *n*).

(e) Procedure `takeOne`: must use procedure `randInt` to randomly take out a bean.

(f) Procedure `getBean`: performs the phrase within the brackets "()" at line 5. It takes as input an array of beans and a bean type (character '`B`' or '`G`') and looks inside the array to find and return a randomly-selected bean that matches the bean type. The found bean is also removed from the array. Parameters order of this function:

```
getBean (char[] beansBag, char beanType)
```

(g) Procedure `updateTin`: performs lines 3-8, which takes as input the `tin` and two beans and updates `tin` accordingly. In addition, this procecure must use `getBean` to obtain a desired bean from `BeansBag`. Parameters order of this function:

```
updateTin (char[] tin, char beanOne, char beanTwo)
```

(h) Procedure `tinGame`: must be modified to use procedure `updateTin`.

Note:

- Use `char` data type for the beans.
- You need to think carefully about the design of each procedure.
- You may ***not*** alter the program structure other than making the changes described above. Reuse the existing procedures when possible.
- You may ***not*** define any other class (static) variables.

3. Assessment and submission requirements

You must submit the assignment exactly as described below:

□ Create a zip-compressed file containing **just the folder of the package** specified in Task 1. You must name the file as follows: **a1_*sid*.zip**, where ***sid*** is your student id.

□ Submit your file to the designated submission box.

□ You must use JDK 8.

IMPORTANT: Failure to name the file as described above will result in an invalid program. In particular, ONLY the **ZIP** format is accepted. Other formats are NOT accepted.