COSC2391 Further Programming
School of Computing Technologies
RMIT University

Assignment 1 –Semester 1 2024

Introduction

You are required to implement a basic Java program using Java SE 17 or later. This assignment is designed to:

- Test your knowledge of basic Java concepts
- Evaluate your ability to design programming logic
- Practice simple object design in Java.

This is an individual assignment. Your final submission is worth 15%; a simple demo in Week 4 is worth a further 3%, giving a total of 18% for Assignment 1

Academic Integrity (more)

The submitted assignment must be **your own work**. No marks will be awarded for any work which is not created by you.

Plagiarism is treated very seriously at RMIT. Plagiarism includes copying code directly from other students (*or enables such copying)*, the internet, the output of AI systems, or other resources without proper reference. Sometimes, students study and work on assignments together and submit similar files which may be regarded as plagiarism. Please note that you should always create your own assignment even if you have very similar ideas. Plagiarism-detection tools will be used to check all submissions. Penalties may be applied in cases of plagiarism.

Burrito King Restaurant

Burrito King sells **burritos**, **fries** and **sodas**. For simplicity we just call them burritos or soda and do not categorize them further in this assignment. The restaurant sells these food items at the following prices: each burrito is sold for $7, a serve of fries for $4 and sodas for $2.50 each. Burritos are cooked and assembled when they are ordered and take 9 minutes to prepare, up to 2 burritos can be prepared at the same time; sodas are poured instantly, and fries are also served instantly if there are any fries in the warming tray; if there are not enough fries to service an order, then they are cooked in batches of 5 serves, which takes 8 minutes. Fries can be prepared at the same time as burritos. If the user orders more serves of fries than what are available in the café, a message should be displayed telling the user that the café needs to cook some more; 5 serves will be prepared, the number of serves the user needs are provided to them, and the remainder are retained for the next order.

Suppose you are a staff working at the restaurant and use the following application to take customer orders. At the end of the day, you can produce sales reports for Total sales (in dollars), sales done for each item, and how many servings of fries remain (they'll need to be thrown out as waste). You can also update the selling price of items at any time.

Assumptions.

1. All prices as per Australian currency denominations.
2. Correct input can be assumed in this assignment. Error handling is not required in PartA.

Part A

Implement the above scenario **without** necessarily using object-oriented design at this stage. The key focus is to use **appropriate data structures** and **algorithm design**. (Note: you **may** skip Part A and go straight to Part B with the use of OO. Part A is designed to let you focus on the basic functionalities.)

The suggested menu is:

```
   a) Order
   b) Show sales report
   c) Update prices
   d) Exit
```

The first option "Order" has a sequence of actions:

```
  > Select the food item
  > Enter amount
  > Show the returned change
```

The following is a sample interaction with the program. Text in bold and red are user inputs from keyboard.

```
================================================================
Burrito King
================================================================
   a) Order
   b) Show sales report
   c) Update prices
   d) Exit
Please select: a

> Select the food item
   1. Burrito
   2. Fries
   3. Soda
   4. No more
Please select: 1

How many burritos would you like to buy: 3


> Select the food item
   1. Burrito
   2. Fries
   3. Soda
   4. No more
Please select: 2

How many serves of fries would you like to buy: 2
Cooking fries; please be patient
3 serves of fries left for next order


> Select the food item
   1. Burrito
   2. Fries
   3. Soda
   4. No more
Please select: 4
Total for 3 Burritos and 2 fries is $29.
Please enter money: 20
Sorry, that's not enough to pay for order

Please enter money: 30
Change returned $1
Time taken: 18 minutes

================================================================
Burrito King
================================================================
   a) Order
   b) Show sales report
   c) Update prices
   d) Exit
Please select: b
```

```
Unsold Serves of Fries: 3

Total Sales:
Burritos:3      $21.00
Fries:    2      $8.00
Soda:     0      $0.00
-------------------------------------------
          5     $29.00
-------------------------------------------

================================================================
Burrito King
================================================================
   a) Order
   b) Show sales report
   c) Update prices
   d) Exit
Please select: c

> Select the food item to update the price
   1. Burrito
   2. Fries
   3. Soda
   4. Exit
Please select: 1
Please enter new price: 5
The unit price of burrito is updated to $5.


================================================================
Burrito King
================================================================
   a) Order
   b) Show sales report
   c) Update prices
   d) Exit
Please select: d
Bye Bye.
```

Part B

The aim of Part B is to introduce objects. In Part B you have to incorporate object concepts such as Restaurant, FoodItem in the above scenario. The functionality of the driver (main) class would remain the same except for the addition of allowing users to order ***meals*** (see below). You have to organize the data and the functionality into required classes using object oriented concepts. For example, in an object oriented program the price information should reside in the corresponding food item object rather than the main program.

The new program should have the exact same functionalities as specified in Part A except that a user can now also order a ***meal***, which is 1 burrito, 1 serve of fries and 1 soda, at a price with a discount of $3: add the option of ordering a meal (or multiple meals) to the menu, but everything else remains the same, including reporting sales of individual items.

General Requirements

- Use appropriate OO design concepts, e.g., polymorphism, where appropriate
- Your program should validate user inputs. Your programs should not crash or display unexpected data upon invalid input.