**Overview**

**GroceryCo** wants to develop a checkout system. Their pricing schemes are defined such as “apples cost 75 cents each; three apples cost $2.00”

Implement a checkout system which handles their pricing schemes.

Details

Write a console program which takes as input an unsorted list of single items from a shopping cart and performs a “check-out”: printing an itemized receipt and total price.

Item prices can be defined in several ways:

1. An item may be priced individually.

For example, the 'Apple' item might cost 75 cents.

2. Item prices may also be based on quantity purchased.

For example, as a promotion, you may buy three 'Apples' for $2.00.

3. Other promotions could be “buy one get one free” or “buy one, get one for 50% off”

The checkout should be able to accept items in any order and calculate the price appropriately.

For example, an Apple might be scanned, followed by 2 Oranges, and then an Apple again.

Prices and promotions often change at the last minute. Each time a checkout transaction begins, the system should accept a set of pricing rules for all items. Note that there may be multiple pricing rules.

Submit your solution via a GitHub repository. Include a README with any necessary information.

The solution may use third party libraries but must be supplied.

Submissions are due three (3) days from receipt of this exercise.

Criteria

The solution will be examined from these perspectives:

• good design

• readability

• maintenance

• testing

• operational-ness

• aesthetics

**Assumptions:**

1. The "list of single items" in the *ShoppingCart* is created from a list of products. Therefore a *Product* class is created.

2. To be able to produce the *ShoppingCard*, the **"** *unsorted list of single items* **"** is assumed to contain a "*ProductId*" for each selected item by the shopper.

3. Pricing Rules are implemented in the following manner:

3.a ) During the checkout process, a *PricingRule#* is entered for a selected item/product. *PricingRule*# may conform to some numbering system,

3.b) The *PricingRule* returns the discount in 'set dollar amount' or 'calculated dollar amount'. For example, if a *PricingRule* contains "25% discount", it returns $2.50 for the price of $10 of an item, and another contains "Buy 3 for $2.00" will return $2.00.