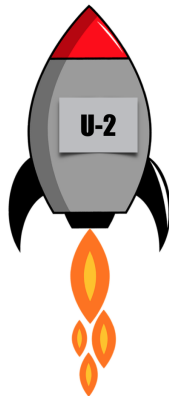
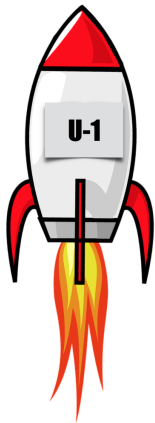


Karabuk University Computer Engineering Department
CME 225E Object Oriented Programming, Fall 2019
Lab # 8 - (Interfaces - Collections)

Space Challenge

In this lab, you will build a simulation that will help us with our mission to Mars!¹. The mission is to send a list of items (habitats, bunkers, food supplies, and rovers) to Mars, but we need to run some simulations first to pick the correct fleet of rockets. We've already designed 2 rocket prototypes, but we need your help to design and run some simulations to help us decide which type to use.



Rocket Specifications

U-1

```
Rocket cost = $100 Million  
Rocket weight = 10 Tonnes  
Max weight (with cargo) = 18 Tonnes  
Chance of launch explosion = 5% * (cargo carried / cargo limit)  
Chance of landing crash = 1% * (cargo carried / cargo limit)
```

U-2

```
Rocket cost = $120 Million  
Rocket weight = 18 Tonnes  
Max weight (with cargo) = 29 Tonnes  
Chance of launch explosion = 4% * (cargo carried / cargo limit)  
Chance of landing crash = 8% * (cargo carried / cargo limit)
```

- U-1:
The U-1 Rocket is light weight, agile and pretty safe, but can only carry a total of 18 tonnes of cargo. It costs \$100 Million to build and weighs 10 tonnes. It has a slim chance of crashing while landing but a bigger chance of exploding when launching, both chances depend on the amount of cargo carried in the rocket.
- U-2:
The U2 Rocket heavier than the U-1 but much safer and can carry a lot more cargo; to a total of 29 tonnes. However, it costs \$120 Million to build and weighs 18 tonnes. It has a greater chance of crashing while landing than while launching, but just like the U-1 both chances depend on the amount of cargo carried.

The Mission

The mission consists of 2 phases:

¹taken from <https://classroom.udacity.com/courses/ud283>

Phase-1

This phase is meant to send building equipment and construction material to help build the colony. 'Phase-1.txt' text file contains the list of all items that we need to send. Each line in the file contains the item name as well as its weight in Kgs.

Phase-2

This phase is meant to send the colony of humans along with some food resources. 'Phase-2.txt' text file contains the list of all items that we need to send. Each line in the file also contains the item name and its weight in Kgs. Your job is to run some simulations and test both rocket types for each phase separately.

Implementation Details

The design

1. Create an **Item** class that includes a **String** name and an **int** weight that will represent an item to be carried by the rockets
2. Create a **SpaceShip** Interface that includes the definitions of these methods:
 - **launch**: a method that returns either **true** or **false** indicating if the launch was successful or if the rocket has crashed.
 - **land**: a method that also returns either **true** or **false** based on the success of the landing.
 - **canCarry**: a method that takes an **Item** as an argument and returns **true** if the rocket can carry such item or **false** if it will exceed the weight limit.
 - **carry**: a method that also takes an **Item** object and updates the current weight of the rocket.
3. Create a class **Rocket** that implements the **SpaceShip** Interface and hence implements all the methods above.
 - **launch** and **land** methods in the **Rocket** class should always return **true**. When U1 and U2 classes extend the Rocket class they will override these methods to return true or false based on the actual probability of each type.
 - **carry** and **canCarry** should be implemented here and will not need to be overridden in the U1 and U2 classes
4. Create classes **U1** and **U2** that extend the Rocket class and override the **land** and **launch** methods to calculate the corresponding chance of exploding and return either **true** or **false** based on a random number using the probability equation for each.

The simulation

Create a **Simulation** class that is responsible for reading item data and filling up the rockets. The Simulation class should include these methods:

- **loadItems**: this method loads all items from a text file and returns an ArrayList of Items:

- Each line in the text file consists of the item name followed by = then its weight in kg. For example:

```
habitat=100000
colony=50000
food=50000
```

- **loadItems** should read the text file line by line and create an Item object for each and then add it to an ArrayList of Items. The method should then return that ArrayList.
- **loadU1**: this method takes the ArrayList of Items returned from **loadItems** and starts creating U1 rockets. It first tries to fill up 1 rocket with as many items as possible before creating a new rocket object and filling that one until all items are loaded. The method then returns the ArrayList of those U1 rockets that are fully loaded.
- **loadU2**: this method also takes the ArrayList of Items and starts creating U2 rockets and filling them with those items the same way as with U1 until all items are loaded. The method then returns the ArrayList of those U2 rockets that are fully loaded.
- **runSimulation**: this method takes an ArrayList of Rockets and calls **launch** and **land** methods for each of the rockets in the ArrayList. Every time a rocket explodes or crashes (i.e if launch or land return false) it will have to send that rocket again. All while keeping track of the total budget required to send each rocket safely to Mars. **runSimulation** then returns the total budget required to send all rockets (including the crashed ones).

Running the simulation

Create a **Main** class with the main method and start running the simulation:

1. Create a **Simulation** object
2. Load Items for **Phase-1** and **Phase-2**
3. Load a fleet of U1 rockets for Phase-1 and then for Phase-2
4. Run the simulation using the fleet of U1 rockets and display the total budget required.
5. Repeat the same for U2 rockets and display the total budget for that.

phase-1.txt

```
building tools=2000
building tools=2000
building tools=2000
building tools=5000
building tools=5000
building tools=2000
building tools=1000
building tools=5000
building tools=6000
shelter equipment=5000
construction equipment=5000
plants=1000
steel=8000
books=1000
water=5000
```

phase-2.txt

```
habitat=3000
colony=5000
food=5000
habitat=3000
colony=5000
food=3000
colony=5000
food=3000
```