

- Put all your work into a directory **hw2** of your Git project and push it every several days.
- Submit a link of your repo in Canvas
- Remember to run git commit regularly if you need

## Question 1

We want to work in the voice mail system in the textbook chapter 2. The current system has few things that can be improved. Your task is to improve the system as follows:

- a- The current user interaction after running the command `java MailSystemTester`:

```
Enter mailbox number followed by #
#
Exception in thread "main" java.lang.NumberFormatException: For
input string: ""
```

The system should print a message to the user “invalid entry” instead of throwing exception and exit the program. The system should allow the user to try again. The same error happens when the user enters a letter instead of a number:

```
Enter mailbox number followed by #
d
#
Exception in thread "main" java.lang.NumberFormatException
```

- b- There is no clear way for the user to enter her/his mailbox. The only way is through tweaking the input. For example:

```
Enter mailbox number followed by #
1
#
You have reached mailbox 1.
Please leave a message now.
Hi there!
# ← instead of hanging up using 'H', press '#' to be able to enter the mailbox of the user
Incorrect passcode. Try again!
← now you can enter your mailbox
```

Modify the interface for the user to enter the mailbox in a clear step. For example, after establishing a connection, the system should start by asking the caller “**To leave a message, press (1), to access your mailbox, press (2)**”. The message should also appear after any hang up ‘H’

- c- If a user hits the wrong key for a menu, just repeat the menu with no further error
- d- If a user tries to send a message and hit '#' instead of 'H', wait for the 'H' without any error messages

## Question 2

- Save the files in this question as a.png, b.png, c.png, etc. If any part has more than one image you can save it as c1.png, c2.png, etc.
- Don't draw it by hand. Use a tool to draw it, as suggested in chapter 2.

Gas Station: Assume that there exist at least the following three classes: Person, Pump, and PayStation. Additional classes may be introduced to clarify the interaction.

- a- Draw a use case diagram for the functionality of a single gas pump at a gas station. Assume that a credit or debit card may be used at the pump, as well as pay cash inside the gas station. Cash may only be used inside of the station. There are three brands of gasoline: low grade, mid grade, and high grade.
- b- Draw a sequence diagram for the following scenario:

“A customer pulls into the gas station to fill up at the gas pump. The customer presses the "pay by credit" button, and fills his tank with low grade and mid grade gasoline (Assume switching from one grade of gasoline to another is a trivial operation). At the end of filling the gas tank the total charge is displayed at the pump, and the customer is asked to swipe his credit card. Realizing that he mistakenly left his credit card at home, the customer proceeds to pay inside the gas station with cash.”

- c- Draw the CRC for each class and the dependency relations between classes