Lab 6 (2) – Revision

1. The pump prices at a petrol station are as follows:

Pump	Type	Price/litre
1	Super	$2.50
2	Unleaded 97	$2.00
3	Unleaded 95	$1.50
4	Diesel	$1.20

   a. Write a program to prompt for a pump number, the litres pumped, and display the price. The program allows for multiple inputs, until user enters 0 when prompted for the pump number. The program should validate pump number and must keep prompting user until he enters 0 to 4 for pump number. E.g.

```
Enter pump no: 1
Litres pumped: 22.5
Please pay $56.25
Enter pump no: 0
(program ends)
```

   Draw a flowchart for this question, before writing the program.

   b. Write a function `getValidInput()` that prompts for a pump no, and displays 'Invalid pump no' if the pump number is not 0,1,2,3,4. Repeat until pump number is valid. The function returns the valid value. Modify part a. to call this function.

   c. Create a list for the price/litre data in the table above. Remove the if…elif…else selection in part a. used for computing the price to pay. Modify your code to use the list.

   d. The station wants to track the number of litres pumped for each pump. Make use of a list to accumulate the litres input for each pump. When the program ends, display the summary as follows:

```
Summary Report
Pump  Type        Total Litres
1     Super       1200
2     Unleaded 97 1059
…
```

   You may introduce other lists such as one to store the petrol types etc.

2. (List of List) Write a program that allows 3 players to guess the total of 3 dice values. The program first gets the player's names, and randomly selects one player to guess first. This is followed by the next player in order of the input of names. The first person to guess the value wins. The program then prompts whether the players wish to play again. If the players do not wish to play, the

the game ends. Display the names of players with the most correct guesses. An example run:

```
Enter player name: A
Enter player name: B
Enter player name: C
(Random chose C to start first)
C, guess the value: 5
Wrong!
A, guess the value: 10
Wrong!
B, guess the value: 7
Wrong!
C, guess the value: 13
Correct!
Play again? (y/n): y
(Random chose A to start first)
A, guess the value: 10
Wrong!
B, guess the value: 15
Correct!
Play again? (y/n): n
Winners: B C
```

3. (Dictionary) Write a program the helps a warehouse manage its inventory.
   a. The program starts by reading an initial inventory list from a file inventory.dat. The content of inventory.dat is as follows:

```
a1 pen 5
a2 mouse 5
a3 keyboard 4
a4 earphone 10
```

   Write a function `readInitialInventoryFromFile()` to read the values from the file and store the data in a dictionary in the following format:

```
inventory = {'a1':['pen', 5], 'a2':['mouse',5] … }
```

   b. Display a menu as follows:

```
Menu
1. Add new inventory
2. Ship inventory
3. Print inventory
0. Quit
Enter option:
```

   Write one or more functions for each of the menu options.

   For option 1, prompt for inventory id, name and qty, and add them to the inventory dictionary. Ensure that the id does not exist in the inventory before adding to the inventory dictionary.

For option 2, prompt for inventory id. If id exists, display the name and qty. Then prompt for the quantity to ship. Make sure the stock level does not go below 0.

For option 3, display the inventory list, indicating a remark ‘Need to reorder’ for items with stock level below 5. Format is as follows:

```
Id Name     Qty   Remarks
a1 pen      5
a2 mouse    5
a3 keyboard 4     Need to reorder
a4 earphone 10
```