This is your third assignment. It is **11%** of your final mark. You need to do it as a group. And you will upload ONE SINGLE FILE PER GROUP.

**Recall the assignment 2 ?!**

**Your assignment 3 is to modularize your solution to the assignment 2.**

**Your challenge is to refactoring your solution so that it uses functions.**

**You decide how to modularize the program.**

**Here are a few hints:**

**- Create functions for each major function**

**- Keep the functions small**

**- Remember the Boss/Worker analogy**

**- Keep the vector declaration in the main function and pass the vector object**

**to any function that requires it**

**DO NOT move the vector object outside main and make it a global variable.**

**- You can start by defining a function that displays the menu**

**- You can then define a function that reads the selection from the user and returns it in uppercase**

**- Create functions for each menu option**

**- Create functions that display the list of numbers, calculates the mean and so forth**

**Take it one function at a time and take your time.**

**Finally, don't forget to use function prototypes!**

**Good luck -- I know you can do it!**

**//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ASSIGNMENT 2 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//**

**This is your second assignment. You need to do it as a group. And you will upload ONE SINGLE FILE PER GROUP.**

**This challenge is about using a collection (list) of integers and allowing the user**

**to select options from a menu to perform operations on the list.**

**Your program should display a menu options to the user as follows:**

**P - Print numbers**

**A - Add a number**

**M - Display mean of the numbers**

**S - Display the smallest number**

**L - Display the largest number**

**Q - Quit**

**Enter your choice:**

**The program should only accept valid choices from the user, both upper and lowercase selections should be allowed.**

**If an illegal choice is made, you should display, "Unknown selection, please try again" and the menu options should be displayed again.**

**If the user enters 'P' or 'p', you should display all of the elements (ints) in the list.**

**If the list is empty you should display "[] - the list is empty"**

**If the list is not empty then all the list element should be displayed inside square brackets separated by a space.**

**For example, [ 1 2 3 4 5 ]**

**If the user enters 'A' or 'a' then you should prompt the user for an integer to add to the list**

**which you will add to the list and then display it was added. For example, if the user enters 5**

**You should display, "5 added".**

**Duplicate list entries are OK ! (for now. check a bit further for added functionalities…)**

**If the user enters 'M' or 'm' you should calculate the mean or average of the elements in the list and display it.**

**If the list is empty you should display, "Unable to calculate the mean - no data"**

**If the user enters 'S' or 's' you should display the smallest element in the list.**

**For example, if the list contains [2 4 5 1], you should display, "The smallest number is 1"**

**If the list is empty you should display, "Unable to determine the smallest number - list is empty"**

**If the user enters 'L' or 'l' you should display the largest element in the list**

**For example, if the list contains [2 4 5 1], you should display, "The largest number is 5"**

**If the list is empty you should display, "Unable to determine the largest number - list is empty"**

**If the user enters 'Q' or 'q' then you should display 'Goodbye" and the program should terminate.**

**Before you begin. Write out the steps you need to take and decide in what order they should be done. PSEUDOCODE**

**Think about what loops you should use as well as what you will use for your selection logic.**

**This exercise can be challenging! It may likely take a few attempts before you complete it -- that's OK!**

**Finally, be sure to test your program as you go and at the end.**

**Hint: Use a vector!**

**Additional functionality if you wish to extend this program.**

**- search for a number in the list and if found display the number of times it occurs in the list**

**- clearing out the list (make it empty again) (Hint: the vector class has a .clear() method)**

**- don't allow duplicate entries**

**- come up with your own ideas! (add up one more functionality. Be creative !!)**

**Good luck!**