

**VGP125 Intro to C# Programming**

**Assignment 1**

Functions in C#

*Estimated hours:​ 4 hours per milestone*

**

Assignment Goals:

* Apply and practice some basic concepts in C# related to logics and functions
* Train and be comfortable with using C# in Visual Studio environments

Assignment Hand-in

* Omnivox
* **Naming Format: VGP125\_A1\_LastName\_FirstName**

Assignment Overview

Please answer the following questions. Make sure that all of them are written in C# using the Visual Studio 2022 IDE. Also, make sure that all of them are written in the Program.cs command, and the functions are created either on top or at the bottom of the script.

1. **Calculator (3 points)**

Ask the player for two numbers (floats) and a basic arithmetic symbol (+, -, \*, /, or %) character. Pass these into a function named **CalculateResult**, and it will return the resulting value of the math equation formed. Printing the returned result, along with the whole equation. Remember in C#, **modulus will work for float numbers**.

***Example:***

*Input first number:* ***50***

*Input second number:* ***40***

*Input math symbol:* ***-***

*Result: 50 – 40 = 10*

1. **Calculate Final Average (except one) (3 points)**

Create a list of grades (float) and then will ask the user to enter any number of grades until they enter a negative number (make sure not to this into the list). Supply this list to a function called **CalculateFinalGrade**. In this function, calculate the average of all the quizzes together, EXCEPT the lowest grade in the list, and then return their final average.

Print the resulting average grade afterwards.

***Example:***

*Input grades:* ***50***

*Input grades:* ***60***

*Input grades:* ***40***

*Input grades:* ***15***

*Input grades:* ***85***

*Input grades:* ***-1***

*The final average is: 58.75*

//15 was removed from the calculation

1. **Remove Guests (4 points)**

Create a void function called RemoveGuests that takes a list of guests. Ask the player to enter names until they enter ‘done’ (don’t include done in the guest list). The function will remove all the guests whose names are longer than 10 characters. Print the guest list before calling the function and after calling it.

***Example:***

*Input guests:* ***Tommy***

*Input guests:* ***Randy***

*Input guests:* ***12characters***

*Input guests:* ***Very Long Name***

*Input guests:* ***Derrick***

*Input guests:* ***done***

*Guest list: Tommy, Randy, 12characters, Very Long name, Derrick*

*Updated guest list: Tommy, Randy, Derrick*

1. **Illegal Items (5 points)**

Ask the player for what items they are bringing into the country, and store them into a list of string. Continue this until the buyer has printed out ‘done’.

Then, pass this list into a function named **CheckContrabands**. Have a pre-made list of strings inside the function, which contains all the items to check for. If any of the brought items matched a name inside the contraband list, display a message displaying which contraband was found.

Lastly, this function will return a bool to check if a contraband has been found or not. If at least one contraband was found, display a message (outside of the function) saying how many of the contrabands were confiscated.

***Example:***

*Input items bringing in the country:* ***Soap***

*Input items bringing in the country:* ***Toys***

*Input items bringing in the country:* ***Drugs***

*Input items bringing in the country:* ***Clothes***

*Input items bringing in the country:* ***done***

*//assume contraband list was {Cigarette, Drugs, Gun, Weed}*

*Contraband found: Drugs*

*1 item has been confiscated.*

1. **Valid IP Address? (5 points)**

Ask the player to enter an IP address as a string. Then create a function called **IsValidIPAddress** that will receive this string and check if the inputted IP address is valid. The function will return true if it’s valid and false if otherwise. If the IP address is invalid, let the player input again. Otherwise, print that it’s valid.

The following conditions have to be true in order for the IP Address to be valid:

* It must be four numbers separated by a dot
* The numbers can only range from 0 to 255.

***Example:***

*Input IP Address:* ***45.67.893.672***

*Invalid IP Address! Please try again.*

*Input IP Address:* ***45/34a33.1***

*Invalid IP Address! Please try again.*

*Input IP Address:* ***45.67.89.100***

*Valid IP Address!*

*//HINT: string.Split() will help you greatly in this question.*