CS2006 C++ Lab Exercises
Week 8

1. Write a program that uses a structure named MovieData to store the following information about a movie:
   Title
   Director
   Year Released
   Running Time (in minutes)
   The program should create two MovieData variables, store values in their members, and pass each one, in turn, to a function that displays the information about the movie in a clearly formatted manner.

2. Write a program that uses a structure to store the following weather data for a particular month:
   Total Rainfall
   High Temperature
   Low Temperature
   Average Temperature
   The program should have an array of 12 structures to hold weather data for an entire year. When the program runs, it should ask the user to enter data for each month. The average temperature should be calculated. Once the data are entered for all the months, the program should calculate and display the average monthly rainfall, the total rainfall for the year, the highest and lowest temperatures for the year (and the months they occurred in), and the average of all the monthly average temperatures.
   Input Validation: Only accept temperatures within the range between -100 and +140 degrees Farenheit.

3. Write a program that keeps track of a speakers' bureau. The program should use a structure to store the following data about a speaker:
   Name
   Telephone Number
   Speaking Topic
   Fee Required
   The program should use an array of at least 10 structures. It should let the user enter data into the array, change the contents of any element, and display all the data stored in the array. The program should have a menu-driven interface.
   Input Validation: When the data for a new speaker is entered, be sure the user enters data for all fields. No negative amounts should be entered for a speaker's fee.

4. Write a program that simulates inventory bins in a warehouse. Each bin holds a number of the same type of parts. The program should use a structure that keeps the following data:
   Description of the part kept in the bin

Number of parts in the bin
The program should have an array of 10 bins, initialized with the following data:

Part Description	Number of Parts in the
binValve	10
Bearing	5
Bushing	15
Coupling	21
Flange	7
Gear	5
Gear Housing	5
Vacuum Gripper	25
Cable	18
Rod	12

The program should have the following functions:
AddParts: a function that increases a specific bin's part count by a specified number
RemoveParts: a function that decreases a specific bin's part count by a specified number.

When the program runs, it should repeat a loop that performs the following steps: The user should see a list of what each bin holds and how many parts are in each bin. The user can choose to either quit the program or select a bin. When a bin is selected, the user can either add parts to it or remove parts from it. The loop then repeats, showing the updated bin data on the screen.

Input Validation: No bin can hold more than 30 parts, so don't let the user add more than a bin can hold. Also, don't accept negative values for the number of parts being added or removed.