**Instructions:**

1. Create a MS VisualStudio solution named **Assignment12**.
2. Create projects according to the assigned problems.
3. Use techniques and experience that you have obtained from previous chapters and this chapter.
4. Code and test your programs. They **MUST** be completed without syntax, logic and run-time errors.
5. Include comments of your name, date, and brief descriptions in all source codes.
6. Compress **Assignment12**folder into **ONE** zipped file.
7. Submit or re-submit your zipped file before its due date&time.
8. int main()  
   {  
      do  
      {  
          switch (menuOption())  
          {  
          case 0: exit(1); break;  
         case 8: Challenge8(); break;  
         case 11: Challenge11(); break;  
         case 12: Challenge12(); break;  
         case 13: Challenge13(); break;   
         case 14: Challenge14(); break;  
         case 15: Challenge15(); break; // 10PTS Extra Credit  
          default: cout << "\t\tERROR - Invalid option. Please re-enter."; break;  
          }  
          cout << "\n";  
          system("pause");  
      } while (true);  
     
       return EXIT\_SUCCESS;  
   }
9. Array/File Functions

Write a function named arrayToFile. The function should accept three arguments: the name of a file, a pointer to an int array, and the size of the array. The function should open the specified file in binary mode, write the contents of the array to the file, and then close the file.

Write another function named fileToArray. This function should accept three arguments: the name of a file, a pointer to an int array, and the size of the array. The function should open the specified file in binary mode, read its contents into the array, and then close the file.

Write a complete program that demonstrates these functions by using the arrayToFile function to write an array to a file, then using the fileToArray function to read the data from the same file. After the data are read from the file into the array, display the array’s contents on the screen.

1. Corporate Sales Data Output

Write a program that uses a structure to store the following data on a company division:

Division Name (such as East, West, North, or South)

Quarter (1, 2, 3, or 4)

Quarterly Sales

The user should be asked for the four quarters’ sales figures for the East, West, North, and South divisions. The data for each quarter for each division should be written to a file.

Input Validation: Do not accept negative numbers for any sales figures.

1. Corporate Sales Data Output

Write a program that uses a structure to store the following data on a company division:

Division Name (such as East, West, North, or South)

Quarter (1, 2, 3, or 4)

Quarterly Sales

The user should be asked for the four quarters’ sales figures for the East, West, North, and South divisions. The data for each quarter for each division should be written to a file.

Input Validation: Do not accept negative numbers for any sales figures.

1. Corporate Sales Data Input

Write a program that reads the data in the file created by the program in Programming Challenge 11. The program should calculate and display the following figures:

Total corporate sales for each quarter

Total yearly sales for each division

Total yearly corporate sales

Average quarterly sales for the divisions

The highest and lowest quarters for the corporation

1. Inventory Program

Write a program that uses a structure to store the following inventory data in a file:

Item Description

Quantity on Hand

Wholesale Cost

Retail Cost

Date Added to Inventory

The program should have a menu that allows the user to perform the following tasks:

Add new records to the file

Display any record in the file

Change any record in the file

Input Validation: The program should not accept quantities, or wholesale or retail costs, less than 0. The program should not accept dates that the programmer determines are unreasonable.