3. You have a voucher for a bakery, which you redeem to buy baked goods of your choice. The value of the voucher is fixed and the total bill of the items selected should not exceed the voucher value. If the voucher has any value left after the purchase, then it cannot be utilized again. Write a program that displays 4 items and their prices as a menu and then reads the value of voucher.
   Write a function, with parameters *itemCode* and *voucherValue* passed by reference, that deducts the price of an item from the voucher's value only if the item's value is less that or equal to the voucher value. Use a loop to invoke the above function as many times as required until the value of the voucher is less than or equal to the cheapest item on the menu.
   Finally, display the calculated sum of the prices of the items selected, and the unutilized balance on the voucher after purchase.

7. Write a program that will read a weight in pounds and ounces and will output the equivalent weight in kilograms and grams. Use at least three functions: one for input, one or more for calculating, and one for output. Include a loop that lets the user repeat this computation for new input values until the user says he or she wants to end the program. There are 2.2046 pounds in a kilogram, 1000 grams in a kilogram, and 16 ounces in a pound.

8. Write a program like that of the previous exercise that converts from kilograms and grams into pounds and ounces. Use functions for the subtasks.

9. (You should do the previous two programming projects before doing this one.) Write a program that combines the functions of the previous two programming projects. The program asks the user if he or she wants to convert from pounds and ounces to kilograms and grams or from kilograms and grams to pounds and ounces. The program then performs the desired conversion. Have the user respond by typing the integer 1 for one type of conversion and 2 for the other. The program reads the user's answer and then executes an *if-else* statement. Each branch of the *if-else* statement will be a function call. The two functions called in the *if-else* statement will have function definitions that are very similar to the programs for the previous two programming projects. Thus, they will be fairly complicated function definitions that call other functions in their function bodies. Include a loop that lets the user repeat this computation for new input values until the user says he or she wants to end the program.

14. Your time machine is capable of going forward in time up to 24 hours. The machine is configured to jump ahead in minutes. To enter the proper number of minutes into your machine, you would like a program that can take a start time and an end time and calculate the difference in minutes between them. The end time will always be within 24 hours of the start time. Use military notation for both the start and end times (e.g. 0000 for midnight and 2359 to represent one minute before midnight).
   Write a function that takes as input a start time and an end time using represented as an int using military notation. The function should return the difference in minutes as an integer. Write a driver program that calls your subroutine with times entered by the user.
   Hint: Be careful of time intervals that start before midnight and end the following day.

17. Given the *scores.txt* file described in Programming Project 4.16, write two additional functions. The first function should be named *getPlayerScore* and take a string parameter as input that is a player's name, and it should return the player's high score stored in the file. If the player's name is not in the file, then the function should return 0. The second function should output whether a player's high score is above average, exactly equal to the average, or below average, where the average is computed from all of the scores in the file. You should design the function appropriately (i.e., determine the function name, parameters, and return values).

    EX: scores.txt (sample)

    A possible sample is shown below where Ronaldo's best score is 10400, Didier's best score is 9800, etc.

```
Ronaldo
10400
Didier
9800
Pele
12300
Kaka
8400
Cristiano
8000
```