Part-2 (40%)

Instructions

1. Carefully review the "Part-2 Output Example" (next section) to see how this program is expected to work
2. Code your solution to Part-2 in the provided "**w3p2.c**" source code file
3. This program can be broken down into a few major logical code sections:
   a) Variable declarations: All variables should be declared together into meaningful groups at the top of the main function
   b) Product data input: Prompt for data describing three (3) products and store user-input to appropriate variables
   c) Display product data: Summarize the product data in a tabular format to help make the data easy to read
   d) Customer preference input (2 times): Prompt for the user's coffee preferences and store user-inputs to appropriate variables
   e) Display summary of results: Apply the customer-input preferences; match how each product meets the needs of the customer defined preferences
   f) Repeat: Repeat step (d) and (e) for another preference scenario
4. Don't delete or modify the provided "***GRAMS_IN_LBS***" variable declaration (you will need this in the conversion from grams to pounds when appropriate).
5. Using the example output as a guide, declare the necessary **nine (9)** variables used to represent the three (3) product data "records".

   Note

   - You must select the **appropriate data type** for each variable based on the type of data that needs to be stored
   - You must use **self-describing variable names** to maximize readability and maintainability of the code

6. Prompting user-input for a **single-character** value can cause unexpected behaviour which you will learn about later in the semester, however, for now use the following ***scanf*** formatting specifier (between the double-quotes) to avoid strange behaviour (notably the **single-space before the percent sign**):

```
scanf(" %c", ...
```

7. Displaying the product data in a tabular format requires the application of some slightly more advanced formatting features (you will learn more about this later in the semester). For now, use the printf statement provided as a comment in the supplied starter w3p2.c file that will look like the below:

```
printf(" 1 |   %d   |   %d    |   %d    | %4d | %6.3lf |   %d   |\n",
```

8. The below table provides the mapping rules you must apply in matching the customer input preferences to each product (Example: if the user prefers "Light" coffee, and the product type is "Light" this would show as true (1) in the summary table result):

Customer Preference		**Product**	
Coffee Strength		**Coffee Type**	
Light	(l or L)	Light	(l or L)
Medium	(m or M)	Medium	(m or M)
Rich	(r or R)	Rich	(r or R)
Daily Servings *(inclusive range)*		**Coffee Package Weight**	
1 to 4		250 g	
5 to 9		500 g	
10 or more		1000 g	
Like Cream with Coffee		**Suggest Serving with Cream**	
Yes	(y or Y)	Yes	(y or Y)
No	(n or N)	No	(n or N)

9. You should not need to declare more than **three (3)** additional variables to store the input values that describe the customer's coffee preferences. The variables will be used in determining how each product "matches" the needs of the customer by comparing the input values to each product (as per above table).
   Reminder All variable declarations MUST be grouped together at the beginning of the "main" function to maximize the management of your program variables (will be easy to find being all in one place)
10. Displaying the preference to product summary results in a tabular format requires the application of some slightly more advanced formatting features (you will learn more about this later in the semester). For now, use the printf statement provided as a comment in the supplied starter w3p2.c file that will look like the below:

```
printf(" 1|      %d        |      %d      |   %d   |\n", ...
```

Note

A) You must code the required ***relational and/or logical expression(s)*** using the appropriate variables as required for each mapped field which will provide the true(1)/false(0) results using the matching rules described in the above table.

B) **You MUST code your logic to work with all combinations** – do not limit your logic so that it only works with the sample data.

C) If you applied the correct logic, the code used to produce the outputs for both customer preference parts should be **IDENTICAL**! 😊

Part-2 Output Example (Note: Use the YELLOW highlighted user-input data for submission)

```
Take a Break - Coffee Shop
==========================

Enter the coffee product information being sold today...

COFFEE-1...
Type ([L]ight,[M]edium,[R]ich): l
Bag weight (g): 250
Best served with cream ([Y]es,[N]o): y

COFFEE-2...
Type ([L]ight,[M]edium,[R]ich): R
Bag weight (g): 500
Best served with cream ([Y]es,[N]o): N

COFFEE-3...
Type ([L]ight,[M]edium,[R]ich): m
Bag weight (g): 1000
Best served with cream ([Y]es,[N]o): n

---+------------------------+---------------+-------+
   |    Coffee              |   Packaged    | Best  |
   |     Type               |  Bag Weight   | Served|
   +------------------------+---------------+ With  |
ID | Light | Medium | Rich  |  (G) | Lbs    | Cream |
---+------------------------+---------------+-------|
 1 |   1   |   0    |   0   |  250 |  0.551 |   1   |
 2 |   0   |   0    |   1   |  500 |  1.102 |   0   |
 3 |   0   |   1    |   0   | 1000 |  2.205 |   0   |

Enter how you like your coffee...

Coffee strength ([L]ight, [M]edium, [R]ich): L
Do you like your coffee with cream ([Y]es,[N]o): y
Typical number of daily servings: 2

The below table shows how your preferences align to the available products:

--------------------+-------------+-------+
  |     Coffee       |  Packaged   | With  |
ID|      Type        | Bag Weight  | Cream |
--+------------------+-------------+-------+
 1|       1          |      1      |   1   |
 2|       0          |      0      |   0   |
 3|       0          |      0      |   0   |

Enter how you like your coffee...

Coffee strength ([L]ight, [M]edium, [R]ich): M
Do you like your coffee with cream ([Y]es,[N]o): n
Typical number of daily servings: 12
```

The below table shows how your preferences align to the available products:

ID	Coffee Type	Packaged Bag Weight	With Cream
1	0	0	0
2	0	0	1
3	1	1	1

Hope you found a product that suits your likes!