COP2220
Project 2 – Modularized Unit Conversion Tool
100 Points

Submission Requirements
- Submit your project folder via the Submit tool provided on the website
    - Follow the project submission guidelines for the class

Design Documentation Requirements

Note: Create a "Design Documents" folder in the project folder to store the design documents.

- Structure chart (Simply recreate the structure chart provided below. **Don't copy and paste**)
- Flowchart of the **withinRange**() function
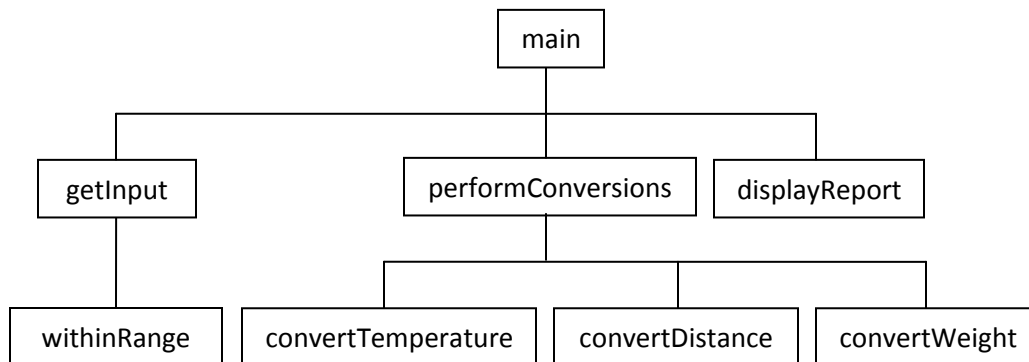- Pseudocode of the **getInput**() function

Design Specification Requirements

Note: Refer to the sample output in the Example Output section below.

1. Modify the program created for Project 1
   Note: You may use the Project 1 Example, if you wish.

    A. The program must conform to the structure chart below
    B. The function declarations provided below must be used


```c
void convertDistance(int feet, double *pMeters, double *pInches);
void convertTemperature(int fahrenheit, double *pCelsius, double *pKelvin);
void convertWeight(int pounds, double *pKilograms, double *pStones);
void displayReport(int fahrenheit, int feet, int pounds, double celsius, double meters,
                   double kilograms, double Kelvin, double Inches, double Stones);
int getInput(int *pFahrenheit, int *pFeet, int *pPounds);
void performConversions(int fahrenheit, int feet, int pounds, double *pCelsius, double
                        *pMeters, double *pKilograms, double *pKelvin, double *pInches,
                        double *pStones);
int withinRange(int input, int minRange, int maxRange);
```

2. Each function must be called from within its parent function only (i.e., the getInput(), performConversions() and displayReport() functions can only be called from within the main() function)
3. After the user enters a value for a given prompt (temperature, distance or weight), perform the following tests
   A. Did the user enter exactly 1 value? (use the **return value** of the **scanf**() function)
      1. If not, display the error message "The entered xyz value is invalid", where xyz is the current prompt (temperature, distance or weight)
   B. Did the user enter a value within the allowable range? (use the **return value** of the **withinRange**() function)
      1. If not, display the error message "The entered value is out of range [x – y]", where x and y represent the low- and high-end of the range, respectively

Note: If any of the above tests fail, the **getInput**() function should **immediately** return control to the **main**() function **after displaying the appropriate error message**, and the **main**() function should exit to the system **without executing the remainder of the program**.

4. Perform the following additional conversions using the provided equations
   A. Fahrenheit to Kelvin $K = (F + 459.67) * 5 / 9$
   B. Feet to inches $I = F * 12$
   C. Pounds to stones $S = P * 0.0714285714$

Note: Three additional variables (of type double) will need to be created to hold the additional converted values. Also, group common conversions within their respective function (i.e., perform the Fahrenheit to Celsius and Kelvin conversions within the convertTemperature() function).

5. Modify the displayed report to include two additional columns to display the results of the additional conversions

Additional Notes
- Use one variable for each entered or converted value (9 variables)
- Do not modify the provided function prototypes
- Do not create any additional functions
- Do not utilize global variables (the 9 variables mentioned above **must be declared within the main() function**)
- Use "flag" variables to represent whether the user entered valid data and whether the entered data was within the allowable range
- Modify the Fahrenheit to Celsius equation to ensure it displays the appropriate decimal value
- Ensure your source code conforms to the commenting standards for the class

Example Output

```
Ima C Student
Modularized Unit Conversion Tool

Enter a Fahrenheit temperature (integer) [0 - 212]: 212
Enter a distance in feet (integer) [0 - 100]: 100
Enter a weight in pounds (integer) [0 - 100]: 100

        Original         Value   Converted to        Value  Converted to        Value
        --------         -----   ------------        -----  ------------        -----
       Fahrenheit          212        Celsius      100.000        Kelvin      373.150
             Feet          100         Meters       30.480        inches     1200.000
           Pounds          100      Kilograms       45.360        stones        7.143
```

```
Ima C Student
Modularized Unit Conversion Tool

Enter a Fahrenheit temperature (integer) [0 - 212]: 213

The entered value is out of range [0 - 212].
```

```
Ima C Student
Modularized Unit Conversion Tool

Please enter a Fahrenheit temperature (integer) [0 - 212]: abc

The entered temperature value is invalid.
```