

COMP1200-C - Assign 03  
Due midnight – Wednesday – February 8  
**Submit assign03.c via Blackboard**

This assignment continues to demonstrate how to approach solving a large problem by solving one smaller part at a time. By solving a smaller part correctly before adding the next, one can keep the number of statements and errors that may result from them to a minimum. The addition requirements demonstrate how an existing problem may change in scope and thus the solution program must be modified. By saving the solution files with an incremental file name, additional versions can easily be saved using subsequent names providing good backup files.

**Before you start writing your program:**

Read the Software Development Plan to plan your solution to the assignment problem.  
Save your assign02b.c as assign03.c

**NOTE: Your submitted file(s) MUST be spelled and cased as instructed.**

*Read all instructions before beginning your work.*

**Program: assign03.c**

A growing number of people are becoming more health conscious watching their diet and exercising more. Statistics are available to help them monitor their progress toward becoming a healthier person. The body mass index (BMI) is a statistical measure which compares a person's weight and height. Though it does not actually measure the percentage of body fat, it is used to estimate a healthy body weight. The following equation is used to calculate body mass index. Note: The mass is the body weight in kilograms and height is in meters. [[http://en.wikipedia.org/wiki/Body\\_mass\\_index](http://en.wikipedia.org/wiki/Body_mass_index)]

In addition to the requirements in the previous assignment, you are ask to determine the classification\* of the computed BMI using the ranges below.

\*According to the National Heart, Lung, and Blood Institute classifications, body mass index is used to determine whether an individual is overweight or obese:

normal            < 25  
>= 25 overweight < 30  
>= 30 obese

You have discovered equations to calculate the Ideal Body Weight (IBW) and want to add this feature to your program. The following equations are used to compute the IBW, where height is in **inches** and the IBW is in **kilograms**. The user will need to specify the gender by entering a character notation, like **m** or **f**.

IBW (men)        = 50.0 + 2.3 \* (height - 60)  
IBW (women) = 45.5 + 2.3 \* (height - 60)

**Additional Problem Constants:**

None.

**Algorithm:**

**Note: the sections headers.**

- Prompt the user to enter a value for *weight in pounds* and *height in inches*.

**Additional Problem Inputs:**

gender

- ☐ BMI
  - Compute conversions.
  - Calculate *BMI*.
  - Display *BMI*.
  - Display *BMI classification*

**Additional Problem Outputs:**

BMI classification  
IBW

- ☐ Target BMI and weight
  - Prompt the user to enter a value for *BMI*.
  - Compute *weight*.
  - Compute conversions.
  - Display *weight*.

**Additional Other variables:**

none

- ☐ IBW
  - Prompt user to enter the *gender* (F/f or M/m).
  - Compute *IBW* for given *height* and *gender*.
  - Display *IBW*.

### Sample Input/Output:

Enter the height in inches: 72  
Enter the weight in pounds: 165

The BMI is 22.38  
BMI Classification: Normal

Enter the target BMI: 25  
The target weight is 184.33 pounds.

Is the person a female or male? Enter F or M: M  
The ideal weight is 171.08 pounds.

New commands  
if...else  
if...else if...else

### General Instructions:

- ☐ Insert comments at the top and throughout each file
  - Include the follow comments at the beginning of this (and ALL) files.
    - // your name
    - // assignment number
    - // date you completed the assignment
    - // **statement(s) about collaboration**
    - // a short narrative about what the file does
  - Use the algorithm as comments throughout each file.
  - **Add section headers. See Standards for Documentation of C programs.**
- ☐ Use descriptive variable names.
- ☐ Use Sample Input/Output as a guide.
- ☐ Use descriptive labeling with output.
- ☐ Divide you solution program code into sections as noted in the algorithm.  
Use section comments as well as the algorithm step comments.
- ☐ Assign01b.c used **scanf** for a character; notice that there is a space in front of **%c**.

-7 points per file for absence of any  
of these required comments at the top

### Submit via Blackboard:

assign03.c      C program file

**NOTE: Your submitted file(s) MUST  
be spelled and cased as instructed.**

### CAUTION!!!

