

BMI Calculator – Assignment 1

Due: February 26, 2013

Spring 2013 Computer Science 411

Mobile Device Application Programming

Instructions: Please submit your solution to the following exercise by 23:59 on February 26, 2013 via «<http://dropbox.ecs.fullerton.edu/>». Your submission must, at a minimum, include a plain ASCII text file README, all necessary source files, libraries, and build configuration files to allow the submission to be built and run independently by the instructor. **All files** must include a header identifying the author, author's contact information, and a brief description of the file. The README file must describe any bugs and features along with a description of what was or was not completed in the solution. The README must also describe the use of the program and if there are any external dependencies. **Do not include any object files, binary executables, or other superfluous files.**

Place your submission in a folder and name it *lastname.firstname_asgt1*. **Zip** the folder and submit the zipped folder, named *lastname.firstname_asgt1.zip*, via «<http://dropbox.ecs.fullerton.edu/>». For example, if your name was Tilly Titan, then your file would be named *Titan.Tilly_asgt1.zip*. **Do not use any other archiving method other than zip.**

Failure to follow these instructions will detrimentally effect your assignment's score.

Plagiarism and academic dishonesty is not tolerated. Correctly and properly attribute all third party material and references.

Do not make calls to `system()`, `exec()` or similar system calls which trigger the operating system to execute an external program.

Assignment

Objective: The objective of this assignment is to gain familiarity with Xcode 4's interface building features and how user interface elements communicate to user-defined code. This assignment results in creating an iPhone application called *BMI* which allows a user to enter a subject's height and weight to calculate and visualize the subject's body mass index.

Rubric: The assignment is out of 100 points.

- 60 points for a correctly executing application
- 30 points for compiling with no errors or warnings
- 10 points for using the correct name for files, the program and documenting your code.

Not including a README forfeits 25 points. Your code is expected to be well organized and well documented¹. Points will be deducted for poorly structured or undocumented code.

Prerequisites: In order to complete the exercise successfully, you will need to have the following:

- Apple Xcode

Requirements: Write an iPhone application with a UI similar to the one pictured in Figure 1. The *Calculate* button takes the input from the two text fields and reports the results back to the user via a label. The result of the calculation is used to look up from Table 1 advice regarding the subject's BMI. Additionally, a graphical representation of the calculated results should be shown somewhere in the view. This may be photographs, cartoon drawings, a chart, a graph, etc.

You are encouraged to be creative with the UI and how you communicate the results back to the user. Experiment with sound, animation, images, etc. to give the application a look and feel that you will be proud of.

¹Well organized and well documented means sufficient comments to explain what all the functions do and even what is going on within a function

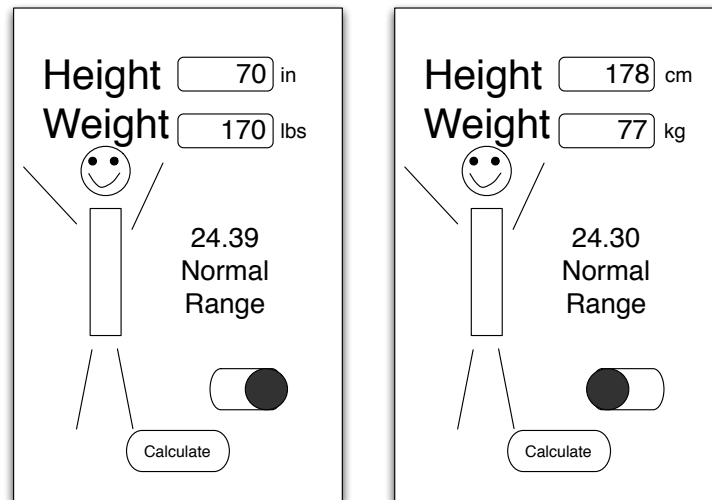


Figure 1: Mock-up of the applications user interface. US customary weights and measures are used in the left and metric weights and measures are used in the right. The toggle switch flips between the two states.

Category	BMI range ($\frac{kg}{m^2}$)
Severe Thinness	<16.00
Moderate Thinness	16.00–16.99
Mild Thinness	17.00–18.49
Normal Range	18.50–24.99
Overweight	25.00–29.99
Obese Class I (Moderate)	30.00–34.99
Obese Class II (Severe)	35.00–39.99
Obese Class III (Very Severe)	>39.99

Table 1: Table adapted from World Health Organization guidelines.