Things to think about but NOT implement in your lab:

1) The automatic grader performs hundreds of tests of your program. In what way is the automatic grader superior to manual testing using **`scanf`** and **`printf`**?

2) This program demonstrates an important concept for designing robust software. In particular, if the input values were always between 3 and 20, you might be tempted to not check for it. Not checking for valid inputs is a common way software crashes or exhibits a security flaw. Think about what it means if the input was supposed to be between 3 and 20, but it is not.

Portions of this chapter were reprinted with approval from Embedded Systems: Introduction to ARM Cortex-M Microcontrollers, 2013, ISBN: 978-1477508992. For more information on this book, see **http://users.ece.utexas.edu/~valvano/arm/outline1.htm** (http://users.ece.utexas.edu/~valvano/arm/outline1.htm)

Specific reading relevant to Chapter 5:
Book Volume 1 Sections 1.8, 1.9, 2.8, 5.1, 5.3

If you are having trouble understanding the C code in this book , we suggest you read the available free sections of on the **Zyante** site
1) Go to http://utedxfall13.zyante.com (http://utedxfall13.zyante.com)
2) Click register in the upper right (do not click “Subscribe for full access”)
3) Registering is free but subsequent visits to this site you will **Login**.

Reading Zyante is optional and not a formal part of this class.