CS303 Programming Assignment #1: "Hello World!"

Out:	Jan 16, 2013. **Due**: Jan 23, 2013 by 7:00 pm on the CS303 Moodle site.
Total points:	100. approximately 20% of the total homework grade.
Relevant lectures:	Lecture 2 introduces command-line parameters in C.
Textbook:	Page 86 has an example of using command-line parameters.

You must write a C program which has the following features (80 points):

1. It is invoked from the command line using at least two and no more than five command-line parameters. Note that the name of the executable counts as a parameter.
2. If an incorrect number of parameters is used, it should print an error message and return the EXIT_FAILURE.
3. If the correct number of parameters is used, it should:
   a. Print the number of parameters to the console
   b. Print all parameters from argv[1] up to argv[4] to the console.
   c. Convert all parameters from argv[1] up to argv[4] into BDF units and display the result. The BDF unit is "Bags of Dog Food" where 30 lbs = 1 BDF.

These are the tests I will use to grade your submitted program. Note that argv[1] - argv[4] will only be integer values.

Test 1: Does the program handle two command-line parameters? (20 points)

If I type this:	Your program should print this:
`$ ./a.out 30`	`2 parameters.` `30 lbs = 1.00 BDF`

Test 2: Does the program handle three command-line parameters? (10 points)

If I type this:	Your program should print this:

`$ ./a.out 30 756`	`3 parameters.`
	`30 lbs = 1.00 BDF`
	`756 lbs = 25.20 BDF`

Test 3: Does the program handle four command-line parameters? (10 points)

If I type this:	Your program should print this:

`$ ./a.out 1 2 3`	`4 parameters.`
	`1 lbs = 0.03 BDF`
	`2 lbs = 0.07 BDF`
	`3 lbs = 0.10 BDF`

Test 4: Does the program handle five command-line parameters? (10 points)

If I type this:	Your program should print this:

`$ ./a.out 1 2 3 4`	`5 parameters.`
	`1 lbs = 0.03 BDF`
	`2 lbs = 0.07 BDF`
	`3 lbs = 0.10 BDF`
	`4 lbs = 0.13 BDF`

Test 5: Does the program handle one command-line parameters? (15 points)

If I type this:	Your program should print this:
`$ ./a.out`	`Incorrect number of parameters.`

Test 6: Does the program handle six command-line parameters? (15 points)

If I type this:	Your program should print this:
`$ ./a.out 1 2 3 4 5`	`Incorrect number of parameters.`

To receive 100/100 points on this assignment, you must utilize good programming practice (20 points):

- Always submit your assignment in a zipfile with your username in the zipfile name.
- Always give variables meaningful names.
- Use #define preprocessor directives for constant values; you may not necessarily need constant values for your solution.
- Avoid redundant code.
- Document code with useful comments. See example program on website for five pieces of guidance on how to adequately document code with comments.

To achieve maximum points on your submission, consider using this submission checklist before submitting your program to the Moodle course website:

___ "The name of my program source is main.c."
___ "My program compiles successfully."
___ "I ran all the tests (see above) to make sure my program executes correctly."
___ "I followed the five pieces of guidance on commenting programs."
___ "I compressed my source *file* into a zipfile named with my username, e.g., crenshaw13.zip"
___ "I did **not** compress my source file using .rar, .z7, or some other proprietary compression program."
___ "I did **not** compress a DIRECTORY of files."
___ "I uploaded my zipfile to Moodle."