Assignment 2 - Morse Code

21

Background

IT'S THE END OF THE WORLD! In the aftermath of the apocalypse, the only form of communication is the telegraph and you are now the communications expert for the people from Vault 666. However, you know little to nothing about Morse Code and can't learn it to save your life. So, you decide to make a program to help you out!

Figure 1 - Solution Example

Task

1. You are developing a console application that will allow the user to enter in one or more words separated by a space.
2. The program will convert each letter to the corresponding Morse Code equivalent. (e.g. 'H' = '….', 'E' = '.', etc)
3. The program only needs to convert the **upper-case** alphabet; lower-case letters, digits, punctuation and other characters don't need to be converted and you can assume they won't be typed in.

Notes

- Figure 1 is an example of what your program could look like when completed.
- Submit the completed assignment to the appropriate BrightSpace dropbox **before** the due date.
- Refer to the attached rubric to ensure you are completing all the required elements of the assignment.

Criteria	Unsatisfactory	Acceptable	Good	Exceptional	Marks
	0	1	2	3	
Input & Output	- the wrong information was selected for input or nothing was input - the output was unclear and currency was not formatted	- some information was correctly input - all information is output, but it needs to be formatted better	- input is handled - output is clear and appealing, but a small change would improve it	- user input is correctly handled - output to user is very clear and visual appealing	
Variables & Constants	- all variables use the wrong data type - very poor names were used - no naming convention was used	- some data types were chosen well - a few variables were well-named - a naming convention was not well used	- most data types were chosen well - some variables were well-named - naming convention was mostly followed	- most appropriate data types used - well-named - uses appropriate naming conventions	
Morse Code	- characters are not converted correctly to Morse Code - too many errors in Morse Code exist	- some characters are converted correctly to Morse Code - a few errors exist	- most characters are converted correctly to Morse Code - a couple of errors exist	- all characters were correctly converted to Morse Code - words are separated by extra spacing	x2
Letter Output	- letters were not output with the matching Morse Code	- some letters were correctly output with the right Morse Code	- most letters were correctly output with the right Morse Code	- each letter is correctly output with its corresponding Morse Code	
Comments	- little to no comments used	- comments are used, some are meaningful and easily understood - some files and functions have headers	- comments are used extensively, most are meaningful and easily understood - most files/ functions have headers	- not over/under commented - comments are meaningful and easily understood - files/functions have headers - program is self-documenting	
Formatting	- was not indented - there were too many deviations in indentation or placement of braces - very difficult to read	- source code formatting was fairly consistent, but contained some inconsistency with whitespace, brackets, etc	- source code formatting was very consistent with respect to whitespace, brace brackets, parentheses, etc	- standard indentation is used throughout without deviation - placement of braces consistent - easy to read	
				Total	**21**