

# Morse Code

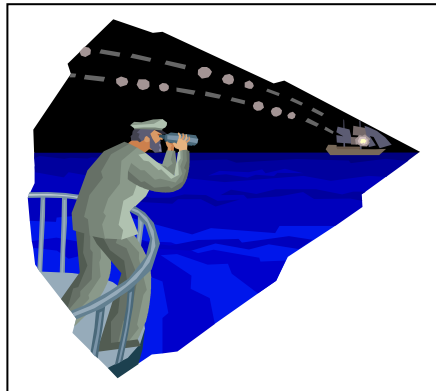
**Program Description:** A C# app is to be created to produce Morse code. The Morse code assigns a series of dots and dashes to each letter of the alphabet, each digit, and a few special characters (such as period, comma, colon, and semicolon). In sound-oriented systems, the dot represents a short sound and the dash represents a long sound. Separation between words is indicated by a space, or, quite simply, the absence of a dot or dash. In a sound-oriented system, a space is indicated by a short period of time during which no sound is transmitted. The international version of the Morse code is stored in the data file **Morse.txt**.

To process the program, the Morse code data file (**Morse.txt**) should be read and stored in memory for fast access to the code. Each letter of the alphabet has a Morse code equivalent. After the code table is stored, the user should be prompted for an English language phrase to be entered from the keyboard. This phrase should be encoded into Morse and displayed. One blank should be used to separate each Morse-coded letter and three blanks should be used to separate each Morse-coded word. The user should be allowed to continue with the process of entering a phrase and having it encoded until a sentinel value (0) is received.

**Input:** The letters and their equivalents are stored in a data file named **Morse.txt**. Input will consist of the **Morse.txt** file and well as the sentences entered from the keyboard. The data file should be read and loaded into memory at the beginning of the program before the sentences can be read and encoded. Each line of the data file contains the letter of the alphabet followed by the code equivalent. The data file should be read and stored as a collection in memory for fast and easy access during the program duration. When a sentence is read from the keyboard, it can be translated from the data stored from the Morse code file. The type of collection that can be used to store the Morse code file may be arrays.

Sample of **Morse.txt**:

```
A  .-
B  -...
C  -.-.
D  -..
E  .
.
.
.
Z  --..
```



**Output:** Output will consist of the display of the original sentence and its Morse code equivalent.

## Requirements:

- Required name for the main program: **Morse.cs**
- You will need to write a separate user-defined class for this program. Please name that **Code.cs**

## Hints:

- In the beginning of the program, read the Morse.txt file and store the data items in parallel ArrayList objects (String data type will be the easiest to work with; however you can use Character if you wish).
- After the file is read and loaded into memory, the user can begin to enter the sentences to be coded.
- Continue to process until the user has finished entering sentences.
- Information on the ArrayList class can be found in the Topic 5 C# Collection Classes

