User’s Documentation

Kewen Deng, 29330440, FIT9133 Assignment 2

1 Instruction

This program contains 5 files(.py) named as decoder_29330440.py, character_29330440.py, word_29330440.py, sentence_29330440.py and main_29330440.py. Each python file gives answer to one single task. Users can have a clear view of the 5 files by following steps:

1) Open main_29330440.py.
2) Click on ‘Run’.
3) Read the Morse Code dictionary in a readable format
4) Import own options of execution.
5) Import Morse Code sequences.
6) Check answers.

It is necessary to mention that users are allowed to input multiple Morse Code sequences not until users would like to terminate (by input nothing and clicking ‘Enter’). Also, each sequence is decoded after input. After all the import, the summary of occurrences about characters, words and sentences is shown according to user's execution options.

Here is the Morse Code table as following:

```
A:01    N:10    0:11111
B:1000  O:111   1:01111
C:1010  P:0110  2:00111
D:100   Q:1101  3:00011
E:0     R:010   4:00001
F:0010  S:000   5:00000
G:110   T:1     6:10000
H:0000  U:001   7:11000
I:00    V:0001  8:11100
J:0111  W:011   9:11110
K:101   X:1001  .:010101
L:0100  Y:1011  ,:110011
M:11    Z:1100  ?:001100
```

2 Limitation

1) Users could only decide the options of different execution (character, word and sentence) at the beginning. Once decided, all the input sequence would execute the same procedures according to the different options given by user.
2) In the analyze of characters, this program would only consider the characters of English letters(A-Z) and numbers(0-9), regardless of punctuations and space.
3) In the analyze of words, this program could just count the occurrences of different spellings of words, regardless of whether the word is a real English word or not. Also, the different tenses and plural numbers of a same word might be considered as different word.

4) In the analyze of sentences, this program would consider that in a whole sentence (end up with '.' or '?'), 2 commas mean 3 clauses. But if the decoded sequence is ended up with ',', 2 commas mean 2 clauses.

e.g Input: One apple, two apples, three apples.

Output: ',':3 '.':1

Input: One apple, two apples,

Output: ',':2 '.':0

3 Assumption

In order to make sure that all the inputs are decodeable and meaningful , here are some assumptions with its different error messages as following:

1) Each sequence should end with punctuation.

```
Please enter a Morse Code sequence. Press Enter to quit.
01*01***001100***001100***11
End without punctuation. This input is invalid!
```

2) Each sequence should contain at least 1 space.

```
Please enter a Morse Code sequence. Press Enter to quit.
011*0000*111*01*11*00*001100
No space exists. This input is invalid!
```

3) Sequences start with * are invalid.

```
Please enter a Morse Code sequence. Press Enter to quit.
***011*0000*111***01*11***00***001100
Start with *. This input is invalid!
```

4) Sequences start with punctuation are invalid.

```
Please enter a Morse Code sequence. Press Enter to quit.
001100***011*0000*111***01*11***00***001100
Start with punctuation. This input is invalid!
```

5) Sequences contain chain of 2 * or more than 3 * are invalid.

```
Please enter a Morse Code sequence. Press Enter to quit.
01**001100
Chain of 2 * or more than 3 *. This input is invalid!
```

6) Sequences contain successive punctuations are invalid.

```
Please enter a Morse Code sequence. Press Enter to quit.
01*01***001100***001100***110011
Chain of more than 1 punctuation. This input is invalid!
```

7) Sequences contain any part of undecodable characters are invalid.

```
Please enter a Morse Code sequence. Press Enter to quit.
01*01*000000000000000***001100
Contains of undecodable message. This input is invalid!
```

4 Demonstration

In order to further illustrate how to run the program, here are some demonstrations simply to show the user interface:

1) Options of executions.

A menu with options allowing the user to select which level of analysis is intended. You could enter either T (execute) or F (not execute).

```
Count character or not? Enter T or F : T

Count word or not? Enter T or F : T

Count sentence or not? Enter T or F : T
```

2) Normal input.

An acceptable input with its decoded sequence.

```
Please enter a Morse Code sequence. Press Enter to quit.
01***01*01***0*0***010101***01***01*01***0*0***010101
A AA EE . A AA EE .
```

3) Execution of character.

If you entered T as the option of character, the output should be like following:

```
The occurrences for each letters are:
A:6
E:4
```

4) Execution of word.

If you entered T as the option of word, the output should be like following:

```
The occurrences for each words are:
A:2
AA:2
EE:2
```

5) Execution of sentence.

If you entered T as the option of sentence, the output should be like following:

```
The occurrences for each type of English sentences are:
.:2
,:0
?:0
```

6) None output of execution.

If the execution returns nothing, which means the input contains no letter, word, or sentence, the output should be like following:

```
No letters occurrence.
No words occurrence.
No sentences occurrence.
```