Assignment 7: Wordle

Start Assignment

Due Mar 20 by 11:59pm **Points** 100 **Submitting** a file upload

The **Wordle** (https://www.nytimes.com/games/wordle/index.html) is a word game. A player will have 6 tries to guess the five-letter word. After each try, a hint about each character will be displayed. For this assignment, we will create our own version of Wordle in Haskell. Not only that, we will also create another program that will help us solve the Wordle.

Wordle (The Game)

A list of five-letter words from Standford is given in the file **sgb-words.txt** (https://canvas.pitt.edu/courses/132828/files/8779856/download?download_frd=1) . For each try, our output will display the hint of each character under it. The '*' character will be used to indicate that the character is in the word and in the correct spot. The character '+' will be used to indicate that the character is in the word but in the wrong spot. Finally, the '-' character will be used to indicates that the character is not in the word. For example,

```
(4) watts
    *+---
```

The above hints state the following:

- `'w'` is in the word and in the correct spot
- `'a'` is in the word but in the wrong spot
- `'t'` and 's' are not in the word

At the beginning, your program should display an example about hints, then display (6), and wait for the player to enter a five-letter word as shown below:

```
Welcome to Wordle by Haskell
============================
(4) watts
    *+---
You of 4 more tries.
'w' is in the word and in the correct spot.
'a' is in the word but in the wrong spot.
't' and 's' are not in the word
```

```
(6)
```

At this point, the cursor should be after (6) and waiting for the player to enter a word. Once a word is enter, display the hints and continue as shown below:

```
(6) asdfi
    ++-+-
(5)
```

If the player correctly guess the word within six tries, congratulate the player and ask whether he/she wants to play again. Otherwise, show the player the correct word and ask whether he/she wants to play again. Here is an example of a run:

```
Welcome to Wordle by Haskell
============================
(4) watts
    *+---
You of 4 more tries.
'w' is in the word and in the correct spot.
'a' is in the word but in the wrong spot.
't' and 's' are not in the word

(6) aeiou
    -++--
(5) steir
    ****-
(4) stein
You Win...
Would you like to play again? (y/n): y
(6) aeiou
    ---+-
(5) motto
    -++*+
(4) osotc
    +-**-
(3) broth
    -****
(2) froth
    -****
(1) troth
    +****
```

```
You lose...The word is "wroth".
Would you like to play again? (y/n): n
```

Wordle (The Helper)

The wordleHelper program will help us narrow down to a small set of words that satisfy hints given by the Wordle (The Game). For example, suppose we enter aeiou and the hints are *-+-+, the hints tell us the following:

1. 'a' is in the word and at the correct spot
2. 'e' is not in the word
3. 'i' is in the word but at the wrong spot
4. 'o' is not in the word
5. 'u' is in the word but at the wrong spot

Starting from the list of words (**sgb-words.txt** (https://canvas.pitt.edu/courses/132828/files/8779856/download?download_frd=1)), you simply perform the following filter:

1. Filter only words that contain 'a' at the first spot
2. Filter only words that do not contain 'e'
3. Filter only words that contain 'i' but not at the third spot
4. Filter only words that do not contain 'o'
5. Filter only words that contain 'u' but not at the fifth spot

The interface will be quite primitive for this program. Ask user to enter a five-letter word and immediately followed by hints. Then reports information about letters that the player entered so far including the number of words left. If the player enter "0", simply display the list of words. Here is an output of a run:

```
There are 5757 words satisfy the following conditions:

Enter a word and hints (0 to show remaining words): aeiou*-+-+
There are 2 words satisfy the following conditions:
 - Do not contain eo
 - Contain 'i' but not at positions [2]
 - Contain 'u' but not at positions [4]
 - Contain 'a' at position 0
Enter a word and hints (0 to show remaining words): 0
["audit","auric"]
There are 2 words satisfy the following conditions:
 - Do not contain eo
 - Contain 'i' but not at positions [2]
 - Contain 'u' but not at positions [4]
```

```
 - Contain 'a' at position 0
Enter a word and hints (0 to show remaining words): audit**-*-
There are 1 words satisfy the following conditions:
 - Do not contain eodt
 - Contain 'i' but not at positions [2]
 - Contain 'u' but not at positions [4]
 - Contain 'i' at position 3
 - Contain 'u' at position 1
 - Contain 'a' at position 0
Enter a word and hints (0 to show remaining words): 0
["auric"]
There are 1 words satisfy the following conditions:
 - Do not contain eodt
 - Contain 'i' but not at positions [2]
 - Contain 'u' but not at positions [4]
 - Contain 'i' at position 3
 - Contain 'u' at position 1
 - Contain 'a' at position 0
Enter a word and hints (0 to show remaining words):
```

Simply press Ctrl-C when done to exit the program.