Introduction to testing day 2 Challenge notes and findings

Test cases and running notes

Test Cases

1

- Test purpose
  - See whether the example in the program description produces the output it says it should
- Text input: a quick brown fox jumps over the lazy dog
- Wrap limit: 26
- Expected output:

```
a quick brown fox jumps
over the lazy dog
--END OF FILE--
```

Notes -

- works as expected

2

- Test purpose
  - using a smaller number and string - one that would break up the words and create many broken lines
- Text input: hello my name is Adam
- Wrap limit: 2
- Expected output:

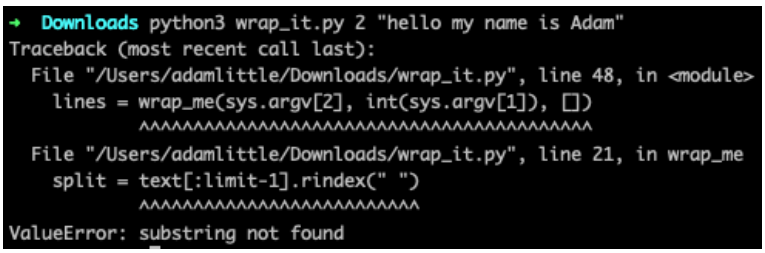
(similar to the below unaware of exact wrapping implementation - from the example it seems likely that the wrapping rounds down to the last full word the first actual printed line is 23 CHAR, but 26 does not include another full word)

```
he
ll
o
my
 n
am
e
is
ad
am
--END OF FILE--
```

alternatively it may just clip the word it is within if the alternative would be 0

```
hello
my
name
is
Adam
--END OF FILE--
```

Notes -



- error given
- not compensated for with an error message or a catch of some kind

---

3

- Test purpose
  - CHAR limit over the length but with no space to seperate by
- Text input: aquickbrownfoxjumpsoverthelazydog
- Wrap limit: 26
- Expected output:

```
aquickbrownfoxjumpsoverthe
lazydog
--END OF FILE--
```

Or an error/exception asking for spaces

Notes -

- same error - different input
- not account for this kind of input with some sort of return value
    - eg printing - you must have spaces!

4

- Test purpose
    - to see how the program will deal with a word that is evenly split along the CHAR limit
- Text input: hello adam smith
- Wrap limit: 8
- Expected output:

```
hello
adam
smith
--END OF FILE--
```

Notes -

- worked as expected

5

- Test purpose
    - See whether the example in the program description produces the output it says it should
- Input: 1 2345 678910 111213141516 1718
    - Input: "1 2345 678910 111213141516 1718"

- Wrap limit: 26
- Expected output:

```
invalid entry
```

Notes -

- when entering the input as multiple 'int' separated by spaces as show
    - output:

```
1
END OF FILE
```

- when entering the input as multiple 'int' separated by spaces as show
    - output:

```
1 2345 678910 111213141516
1718
END OF FILE
```
