CSCA08H Worksheet: Files

1. We have a spreadsheet file that we've opened and assigned to `budget_file`:

```
budget_file = open('budgie_budget.csv')
```

Consider each piece of code on the left and fill in the blank next to each description on the right with the code fragment ((a), (b), (c), (d), or none) that it describes.

(a)
```
for line in budget_file:
    print(line)
```

(b)
```
line = budget_file.readline()
for line in budget_file:
    print(line)
```

(c)
```
for line in budget_file:
    print(line)
    budget_file.readline()
```

(d)
```
print(budget_file.readlines()[0])
```

(1) prints only the first line (d)

(2) prints every line except the first (b)

(3) prints every line twice none

(4) prints all lines (a)

(5) prints every second line (c)

2. Consider this code:

```
budget_file = open('budgie_budget.txt', 'w')
budget_file.write('Seed: $10/month')
budget_file.write('Cage: $50')
budget_file.close()
```

What will the contents of `budgie_budget.txt` look like after this code is run?

(a)
```
'Seed: $10/month'
'Cage: $50'
```

(b)
```
Seed: $10/month Cage: $50
```

(c)
```
Cage: $50
```

(d)
```
Seed: $10/month
Cage: $50
```

(e)
```
Seed: $10/monthCage: $50
```

(f)
```
'Seed: $10/month''Cage: $50'
```

3. Many Unix-like systems (like OSX and the Teaching Labs) have a file of correctly spelled words. See below some of those words (the file contains both capitalised and lowercase words).

```
Zworykin
Zyrtec
Zyrtec's
a
aardvark
aardvarks
abaci
aback
```

Complete the function definition on the next page:

```
def is_correct(dict_file: TextIO, word: str) -> bool:
    """Return True if and only if word is in dict_file.
    Note: current position in dict_file will change as a result of the call.

    Pre: word is not empty
         dict_file is open for reading,
                   current position at the beginning of the file

    >>> with open('dictionary.txt') as dict_file:
    ...     is_correct(dict_file, 'Zyrtec')
    True
    >>> with open('dictionary.txt') as dict_file:
    ...     is_correct(dict_file, 'lolz')
    False
    """
```

4. Complete the definition of the following function:

```
def write_ascii_triangle(outfile: TextIO, block: str, sidelength: int) -> None:
    """Write an ascii isosceles right triangle using block that is sidelength
    characters wide and high to outfile. The right angle should be in the
    upper-left corner.

    Precondition: len(block) == 1
                  sidelength >= 0
                  outfile is open for writing

    For example, given block="@" and sidelength=4, the following
    should be written to the file:

    @@@@
    @@@
    @@
    @

    """
```