CSC108H Worksheet: Files

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

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

   Consider these code fragments:

   (a)
   ```
   for line in f:
       print(line)
   ```
   (b)
   ```
   f.readline()
   for line in f:
       print(line)
   ```
   (c)
   ```
   for line in f:
       print(line)
       f.readline()
   ```
   (d)
   ```
   print(f.readlines()[0])
   ```

   Fill in the blank next to each description below with the code fragment from above, (a), (b), (c) or (d), that it describes.

   (1) prints only the first line ______________

   (2) prints every line except the first ______________

   (3) prints all lines ______________

   (4) prints every second line ______________

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

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

   (e) `Cage: $50`

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

CSC108H Worksheet: Files

3. Many Unix-like systems (including CDF) have a dictionary of correctly spelled words in a file. On CDF, here is the path to the file: `/etc/dictionaries-common/words`. See below some of those words (the file contains both capitalized and lowercase words); complete the function on the right, where `dictionary` refers to a file that has been opened for reading:

```
Zworykin   |  def is_correct(dictionary, word):
Zyrtec     |      """ (file open for reading, str) -> bool
Zyrtec's   |
a          |      Return True iff word is a correctly-spelled word in dictionary.
aardvark   |
aardvarks  |      >>> dict_file = open('dictionary.txt')
abaci      |      >>> is_correct(dict_file, 'Zyrtec')
aback      |      True
           |      >>> dict_file.close()
           |      >>> dict_file = open('dictionary.txt')
           |      >>> is_correct(dict_file, 'lolz')
           |      False
           |      >>> dict_file.close()
           |      """
```

4. Complete the following function:

```
def write_ascii_triangle(outfile, block, sidelength):
    """ (file open for writing, str, int) -> NoneType

    Precondition: len(block) == 1

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

    @@@@
    @@@
    @@
    @
    """
```