Answer Key: CIS 166 Final Exam, Version 1, Spring 2014

1. What will the following code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
print("Two of them are",days[0], days[-1])
result = ""
for i in range(len(days[0])):
    if i > 2:
        result = result + days[0][i]
print("My favorite", result, "is Saturday.")
```

Answer Key:

```
There are 3 fun days in a week
Two of them are Friday Sunday
My favorite day is Saturday.
```

2. Define a Python function named `calculate_tax` which accepts one parameter, `income`, and returns the income tax. Income is taxed according to the following rule: the first $200,000 is taxed at 25% and any remaining income is taxed at 50%. For example, `calculate_tax(100000)` should return $100,000 \times 0.25 = 25,000$, while `calculate_tax(300000)` should return $200,000 \times 0.25 + 100,000 \times 0.5 = 100,000$.

Answer Key:

```
def calculate_tax(income):
    if income < 200000:
        tax = income * .25
    else:
        tax = 200000 * .25 + (income - 200000) * .5
    return tax
```

3. Complete the following program. That is, write the functions `getInputs()`, `countWord()`, `average()`, and `printSummary()`:

```
def main():
    fname, word = getInputs()   #get the file name and word to be searched
    infile = open(fname, "r")   #open the file for reading
    resultList = list()         #initialize result list to empty list

    for line in infile:
        num = countWord(line, word) #return the number of
                                    #times word occurs in line
        resultList.append(num)

    a = average(resultList)     #compute the average number of
                                #times word appears per line
    printSummary(word, a)       #print the average (including explanation)
```

Answer Key:

```
def getInputs():
    fname = input('Enter file name: ')
    word = input('Enter word: ')
    return fname, word

def countWord(line, word):
    return (line.count(word))

def average(l):
    total = 0
    for i in l:
        total = total + i
    return total/len(l)

def printSummary(word, a):
    print("The average number of times per line the word", word)
    print("occurs in the file is", a)
```

4. Given the following function definitions:

```
def bar(n):
    if n <= 8:
        return 1
    else:
        return 0

def foo(l):
    n = bar(l[-1])
    return l[n]
```

(a) What does `foo([1,2,3,4])` return?

Answer Key: 2

(b) What does `foo([1024,512,256,128])` return?

Answer Key: 1024

5. Given the following code:

```
file = open("numbers.txt")
total = 0
for line in file.readlines():
    for strnum in line.split(","):
        num = int(strnum)
        if num % 2 == 0:
            total = total + num
            print(total)
```

(a) What will the output be for this `numbers.txt`?

numbers.txt:

```
1,2,3,4,5,6
```

Answer Key:

```
2
6
12
```

(b) What will the output be for this `numbers.txt`?

numbers.txt:

```
123456
```

Answer Key:

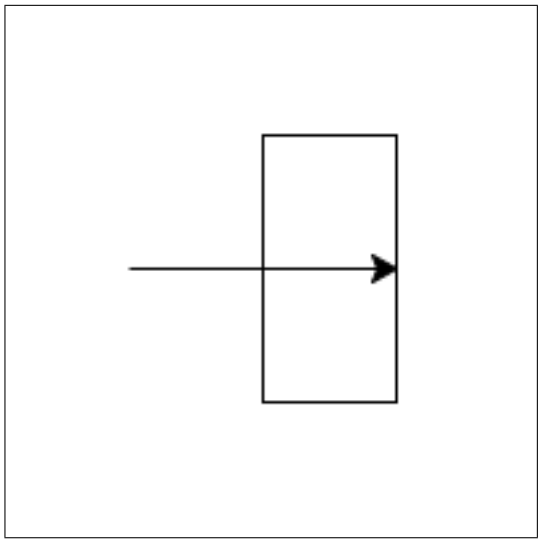
```
123456
```

6. Draw what will be displayed in the graphics window when the following program is executed. Remember to indicate the final position and direction of the turtle at the end of program. (The turtle always points to the right of the screen at the start of the program.)

```
from turtle import *

def mystery(t, n, d):
    for i in range(n):
        if d == 'r':
            t.right(360/n)
        else:
            t.left(360/n)
        t.forward(50)

def draw(t, n):
    t.forward(100)
    mystery(t, n, 'r')
    mystery(t, n, 'l')


t = Turtle()
draw(t, 4)
```

Graphics Displayed:

Answer Key:



7. Write a **program** that reads in a text file, `infile.txt`, and prints out the lines containing the phrase: `The Amazing Spider Man` (that is, the line must contain all four words in this order):

Answer Key:

```
def main():
    infile = open("infile.txt","r")
    for l in infile:
        if l.find("The Amazing Spider Man") != -1:
            print(l)
    infile.close()
```

8. Write the python code for the algorithms below:

   (a)
   ```
   find(st)
       set index to 0
       set location to -1
       set found to false
       while not found
           if st[index] equals ','
               set location to index
               set found to true
           increment index
       return location
   ```

   Answer Key:

   ```
   def find(st):
       index = 0
       location = -1
       found = False
       while not found
           if st[index] == ',':
               location = index
               found = True
           index = index + 1
       return location
   ```

   (b)
   ```
   getSmaller(ls)
       for each item in ls
           if current item is less than first item in ls
               switch first item and current item in ls
   ```

   Answer Key:

   ```
   def getSmaller(ls):
       for i in range(len(ls)-1):
           if ls[i] < ls[0]:
               ls[i],ls[0] = ls[0],ls[i]
   ```

9. Given the following input file `twitter-followers.dat`, write a `program` the reads in the input file and prints the name of the celebrity followed by the word `Amazing` for each celebrity with over 50 million followers, the word `Sweet` for celebrities with followers between 40 and 50 millions and the word `Good` for all others.

twitter-followers.dat
(Celebrity, Followers in Millions)

```
Katy Perry, 52
Justin Bieber, 51
Barack Obama, 42
Lady Gaga, 41
Taylor Swift, 40
Britney Spears, 37
Rihanna, 35
Justin Timberlake, 32
Ellen DeGeneres, 28
```

Answer Key:

```
def main():
    infile = open("twitter-followers.dat", "r")
    for l in infile:
        words = l.split(",")
        print(words[0],end="\t")
        num = int(words[1])
        if num > 50:
            print("Amazing")
        elif num > 40:
            print("Sweet")
        else:
            print("Good")
    infile.close()
```

10. Given the following code:

```
    def getArray():
        input = [
            "Oppenheimer,Robert",
            "Fermi,Enrico",
            "Feynman,Richard",
            "Teller,Edward",
            "Frisch,Otto",
            ",Zazzles"
            ]
        return input
    def main():
        array = getArray()
        last = ""
        for line in array:
            last = line.split(",")[1]
        print(last)
    main()
```

And given following CSV file labeled `cats.txt`:

```
Oppenheimer,Robert
Fermi,Enrico
Feynman,Richard
Teller,Edward
Frisch,Otto
,Zazzles
```

(a) To make the program easier to update, rewrite the `getArray()` function so that it retrieves the input from the CSV file. The updated program should print the same output when run.

Answer Key:

```
def getArray():
    infile = open("cats.txt","r")
    input = []
    for line in infile:
        input.append(line)
    return input
```

(b) What is the output of the program?

Answer Key:

```
Zazzles
```

Answer Key: CIS 166 Final Exam, Version 2, Spring 2014

1. What will the following code print:

```
s = "marchxoctoberxjanuaryxaugustx"
num = s.count("x")
items = s[:-1].split("x")
result = ""
for item in items:
    print( item.capitalize() )
    result = result + item[0].upper()
print( (result[0:2] + "NTHS!! ") * 3, end="")
```

Answer Key:

```
March
October
January
August
MONTHS!! MONTHS!! MONTHS!!
```

2. Define a Python function named `calculate_tax` which accepts one parameter, `income`, and returns the income tax. Income is taxed according to the following rule: the first $100,000 is taxed at 25% and any remaining income is taxed at 50%. For example, `calculate_tax(80000)` should return $80,000 \times 0.25 = 20,000$, while `calculate_tax(200000)` should return $100,000 \times 0.25 + 100,000 \times 0.5 = 75,000$.

Answer Key:

```
def calculate_tax(income):
    if income < 100000:
        tax = income * .25
    else:
        tax = 100000 * .25 + (income - 100000) * .5
    return tax
```

3. Complete the following program—that is, write the functions `getInputs()`, `countAs()`, `average(l)`, and `printSummary(a)`:

```
def main():
    fname = getInputs()         #get the file name
    infile = open(fname, "r")   #open the file for reading
    resultList = list()         #initialize result list to empty list

    for line in infile:
        num = countAs(line)     #return the number of 'a' and 'A' in line
        resultList.append(num)

    a = average(resultList)     #compute the average number of
                                #times 'a' or 'A' appears per line
    printSummary(a)             #print the average (including explanation)
```

Answer Key:

```
def getInputs():
    fname = input('Enter file name: ')
    return fname

def countAs(line):
    return (line.count('A')+line.count('a'))

def average(l):
    total = 0
    for i in l:
        total = total + i
    return total/len(l)

def printSummary(a):
    print("The average number 'A' or 'a' per line")
    print("in the file is", a)
```

4. Given the following function definitions:

```
def bar(n):
    if n >= 32:
        return 2
    else:
        return 1

def foo(l):
    n = bar(l[2])
    return l[n]
```

   (a) What does `foo([1,2,3,4])` return?

   Answer Key: 3

   (b) What does `foo([1024,512,256,128])` return?

   Answer Key: 512

5. Given the following code:

```
file = open("numbers.txt")
total = 0
for line in file.readlines():
    for strnum in line.split(","):
        num = int(strnum)
        if num % 2 == 0:
            total = total + num
            print(total)
```

(a) What will the output be for this `numbers.txt`?

numbers.txt:

```
10,11,12,13,14
```

Answer Key:

```
10
22
36
```

(b) What will the output be for this `numbers.txt`?

numbers.txt:

```
1011121314
```

Answer Key:

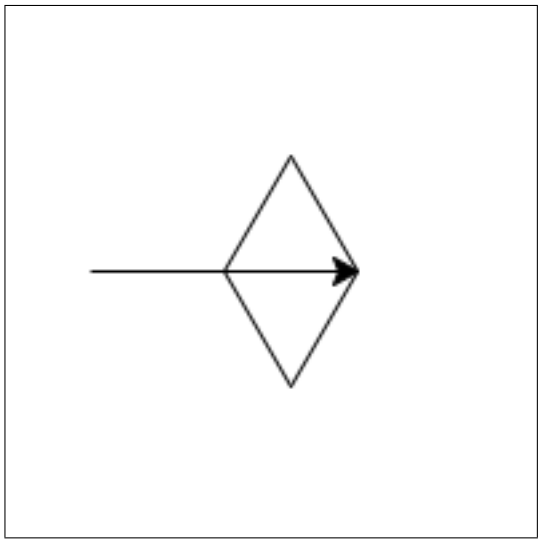
```
1011121314
```

6. Draw what would be displayed in the graphics window when the following program is executed. Remember to indicate the final position and direction of the turtle at the end of program. (The turtle always points to the right of the screen at the start of the program.)

```
from turtle import *

def mystery(t, n, d):
    for i in range(n):
        if d == 'r':
            t.right(360/n)
        else:
            t.left(360/n)
        t.forward(50)

def draw(t, n):
    t.forward(100)
    mystery(t, n, 'r')
    mystery(t, n, 'l')


t = Turtle()
draw(t, 3)
```

Graphics Displayed:

Answer Key:



7. Write a **program** that reads in a text file, `infile.txt`, and replace each line with the word `Awesome` (that is, every line of the `infile.txt` should be `Awesome`), then prints out the total number of lines in the file.

Answer Key:

```
def main():
    infile = open("infile.txt","r")
    lines = infile.readlines()
    numLines = len(lines)
    infile.close()
    outfile = open("infile.txt","w")
    for i in range(numLines):
        print("Awesome",file=outfile)
    print(numLines)
    outfile.close()
```

8. Write the python code for the algorithms below:

   (a)
   ```
   find(st)
       set index to (length of st) - 1
       set location to -1
       set found to false
       while not found
           if st[index] equals ','
               set location to index
               set found to True
           decrement index
       return location
   ```

   Answer Key:

   ```
   def find(st):
       index = len(st) - 1
       location = -1
       found = False
       while not found
           if st[index] == ',':
               location = index
               found = True
           index = index - 1
       return location
   ```

   (b)
   ```
   getBigger(ls)
       for each item in ls
           if current item is greater than first item in ls
               switch first item and current item in ls
   ```

   Answer Key:

   ```
   def getBigger(ls):
       for i in range(len(ls)-1):
           if ls[i] > ls[0]:
               ls[i],ls[0] = ls[0],ls[i]
   ```

9. Given the following input file `facebook-fans.dat`, write a `program` the reads in the input file and prints out name of the celebrity followed by the word `Amazing` for each celebrity with over 85 millions fans, the word `Sweet` for celebrities with fans between 70 and 85 millions and the word `Good` for all others.

facebook-fans.dat (Celebrity, Fans in Millions)

```
Shakira, 87
Rihanna, 86
Eminem, 84
Cristiano Ronaldo, 76
Vin Diesel, 70
Katy Perry, 65
Will Smith, 64
Justin Bieber, 64
Lady Gaga, 63
Linkin Park, 60
```

Answer Key:

```
def main():
    infile = open("facebook-fans.dat", "r")
    for l in infile:
        words = l.split(",")
        print(words[0],end="\t")
        num = int(words[1])
        if num > 85:
            print("Amazing")
        elif num > 70:
            print("Sweet")
        else:
            print("Good")
    infile.close()
```

10. Given the following code:

```
def getArray():
    input = [
        "Stanley,Paul",
        "Simmons,Gene",
        "Singer,Eric",
        "Thayer,Tommy"
        ]
    return input
def main():
    array = getArray()
    last = ""
    for line in array:
        last = line.split(",")[1]
    print(last)
main()
```

And given the following CSV file called `kiss.txt`:

```
Stanley,Paul
Simmons,Gene
```

```
Singer,Eric
Thayer,Tommy
```

(a) To make the program easier to update, rewrite the `getArray()` function so that it retrieves the input from the CSV file. The updated program should print the same output when run. **Answer Key:**

```
def getArray():
    infile = open("kiss.txt","r")
    input = []
    for line in infile:
        input.append(line)
    return input
```

(b) What is the output of the program?

Answer Key:

```
Tommy
```

Answer Key: CIS 166 Final Exam, Version 3, Spring 2014

1. What will the following code print:

```
s = "history.biology.french.trigonometry.science."
num = s.count(".")
subjects = s[:-1].split(".")
print("There are", num, "important subjects in school.")
for item in subjects[:-1]:
    print("Don't know much about", item)
print("But I do know that I love computer " + subjects[4])
```

Answer Key:

```
There are 5 important subjects in school.
Don't know much about history
Don't know much about biology
Don't know much about french
Don't know much about trigonometry
But I do know that I love computer science
```

2. Define a Python function named `calculate_tax` which accepts one parameter, `income`, and returns the income tax. Income is taxed according to the following rule: the first \$50,000 is taxed at 10% and any remaining income is taxed at 20%. For example, `calculate_tax(40000)` should return $40,000 \times 0.1 = 4,000$, while `calculate_tax(100000)` should return $50,000 \times 0.1 + 50,000 \times 0.2 = 15,000$.

Answer Key:

```
def calculate_tax(income):
    if income < 50000:
        tax = income * .10
    else:
        tax = 50000 * .10 + (income - 50000) * .20
    return tax
```

3. Complete the following program that is, write the functions `getInputs()`, `countSpaces()`, `minMax()`, and `printSummary()`:

```
def main():
    fname = getInputs()         #get the file name
    infile = open(fname, "r")   #open the file for reading
    resultList = list()         #initialize result list to empty list

    for line in infile:
        num = countSpaces(line) #return the number of spaces in line
        resultList.append(num)

    m,M = minMax(resultList)     #compute the minimum and maximum spaces per line
    printSummary(m,M)       #print the min and max spaces (including explanation)
```

Answer Key:

```
def getInputs():
    fname = input('Enter file name: ')
    return fname

def countSpaces(line):
    return (line.count(' '))

def minMax(l):
    return min(l),max(l)

def printSummary(m,M):
    print("The minimum number of spaces per line is", m)
    print("The maximum number of spaces per line is", M)
```

4. Given the following function definitions:

```
def bar(n):
    if n < 8:
        return -1
    else:
        return n//2

def foo(l):
    n = bar(l[3])
    return 2*n
```

   (a) What does `foo([1,2,3,4])` return?

   Answer Key: -2

   (b) What does `foo([1024,512,256,128])` return?

   Answer Key: 128

5. Given the following code:

```
file = open("numbers.txt")
total = 0
for line in file.readlines():
    for strnum in line.split(","):
        num = int(strnum)
        if num % 2 == 0:
            print(num)
            total = total + num
print(total)
```

   (a) What will the output be for this `numbers.txt`?

numbers.txt:

```
1,2,3,4,5,6
```

Answer Key:

```
2
4
6
12
```

(b) What will the output be for this `numbers.txt`?

numbers.txt:

```
123456
```

Answer Key:

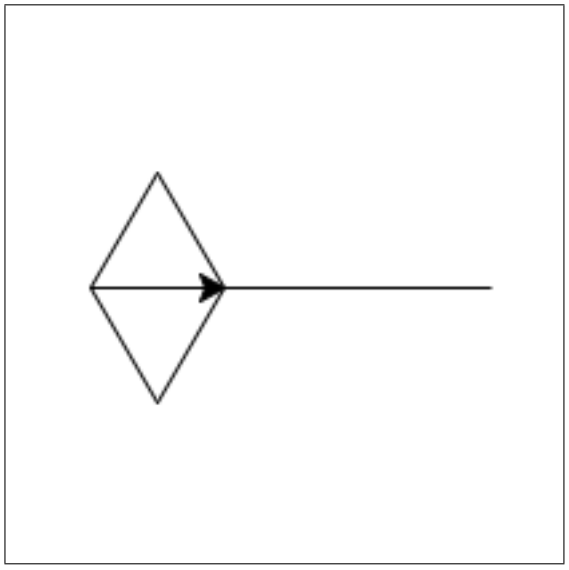
```
123456
123456
```

6. Draw what would be displayed in the graphics window when the following program is executed. Remember to indicate the final position and direction of the turtle at the end of program. (The turtle always points to the right of the screen at the start of the program.)

```
from turtle import *

def mystery(t, n, d):
    for i in range(n):
        if d == 'r':
            t.right(360/n)
        else:
            t.left(360/n)
        t.forward(50)

def draw(t, n):
    t.backward(100)
    mystery(t, n, 'l')
    mystery(t, n, 'r')


t = Turtle()
draw(t, 3)
```

Graphics Displayed:

Answer Key:



7. Write a **program** that reads in a text file, `infile.txt`, and prints out each line surrounded by `'-*-'`.

Answer Key:

```
def main():
    infile = open("infile.txt","r")
    lines = infile.readlines()
    for line in lines:
        print("-*-"+line[:-1]+"-*-")
    infile.close()
```

8. Write the python code for the algorithms below:

   (a)
   ```
   find(st)
       set index to 0
       set location to -1
       set firstFound to false
       set notFound to true
       while notFound and index < length st
           if st[index] equals ',' and firstFound is false
               set firstFound to true
           otherwise, if st[index] equals ','
               set location to index
               set notFound to false
           increment index
       return location
   ```

   Answer Key:

   ```
   def find(st):
       index = 0
       location = -1
       firstFound = False
       notFound = True
       while notFound and index < len(st):
           if st[index] == ',' and firstFound == False:
               firstFound = True
           elif st[index] == ',':
               location = index
               notFound = False
           index = index + 1
       return location
   ```

   (b)
   ```
   getBigger(ls)
       for each item in ls
           if current item is greater than last item in ls
               switch last item and current item in ls
   ```

   Answer Key:

   ```
   def getBigger(ls):
       for i in range(len(ls)-1):
           if ls[i] > ls[-1]:
               ls[i],ls[-1] = ls[-1],ls[i]
   ```

9. Given the following input file `best-selling-albums.dat`, write a `program` the reads in the input file and prints out prints out name of the album followed by the word `Amazing` for each album whose sales

were over 50 millions, the word `Sweet` for sales between 45 and 50 millions and the word `Good` for all others.

best-selling-albums.dat (Album Name, Copies Sold in Millions)

```
Thriller, 65
The Dark Side of the Moon, 45
Eagles Greatest Hits, 42
Back in Black, 40
Saturday Night Fever, 40
Rumours, 40
The Bodyguard, 40
```

Answer Key:

```
def main():
    infile = open("best-selling-albums.dat", "r")
    for l in infile:
        words = l.split(",")
        print(words[0],end="\t")
        num = int(words[1])
        if num > 50:
            print("Amazing")
        elif num > 45:
            print("Sweet")
        else:
            print("Good")
    infile.close()
```

10. Given the following code:

```
    def getArray():
        input = [
            "Targaryen,Daenerys",
            "Baelish,Petyr",
            "Arryn,Lysa",
            "Clegane,Sandor",
            "Stark,Bran"
            ]
        return input
    def main():
        array = getArray()
        last = ""
        for line in array:
            last = line.split(",")[1]
        print(last)
    main()
```

And given the following CSV file labeled `stillalive.txt`:

```
Targaryen,Daenerys
Baelish,Petyr
Arryn,Lysa
Clegane,Sandor
Stark,Bran
```

(a) To make the program easier to update, rewrite the `getArray()` function so that it retrieves the input from the CSV file. The updated program should print the same output when run. **Answer Key:**

```
def getArray():
    infile = open("stillalive.txt","r")
    input = []
    for line in infile:
        input.append(line)
    return input
```

(b) What is the output of the program?

Answer Key:

```
Bran
```

Answer Key: CIS 166 Final Exam, Version 4, Spring 2014

1. What will the following code print:

```
s = "omelettesporridgescerealspancakes"
num = s.count("s")
breakfast = s[:-1].split("s")
print("You have a choice of", num, "options:")
for item in breakfast:
    print(item.capitalize())
print("\nBut I need " + breakfast[0][1] + breakfast[1][1] + breakfast[2][2:4] + "!!!")
```

Answer Key:

```
You have a choice of 4 options:
Omelette
Porridge
Cereal
Pancake

But I need more!!!
```

2. Define a Python function named `calculate_tax` which accepts one parameter, `income`, and returns the income tax. Income is taxed according to the following rule: the first \$500,000 is taxed at 50% and any remaining income is taxed at 75%. For example, `calculate_tax(400000)` should return $400,000 \times 0.5 = 200,000$, while `calculate_tax(600000)` should return $500,000 \times 0.5 + 100,000 \times 0.75 = 325,000$.

Answer Key:

```
def calculate_tax(income):
    if income < 500000:
        tax = income * .50
    else:
        tax = 500000 * .50 + (income - 500000) * .75
    return tax
```

3. Complete the following program that is, write the functions `getInputs()`, `countSpaces()`, `calculate()`, and `printSummary()`:

```
def main():
    fname = getInputs()         #get the file name
    infile = open(fname, "r")   #open the file for reading
    resultList = list()         #initialize result list to empty list

    for line in infile:
        num = countSpaces(line) #return the number of spaces in line
        resultList.append(num)

    n = calculate(resultList)  #compute number of lines with more than 5 spaces
    printSummary(n)       #print the number of long lines (including explanation)
```

Answer Key:

```
def getInputs():
    fname = input('Enter file name: ')
    return fname

def countSpaces(line):
    return (line.count(' '))

def calculate(l):
    total = 0
    for i in l:
        if i > 5:
            total = total + 1
    return total

def printSummary(n):
    print("The number of long lines (more than 5 spaces)")
    print("is", n)
```

4. Given the following function definitions:

```
def bar(n):
    if n >= 8:
        return 8
    else:
        return n*2

def foo(l):
    n = bar(l[1])
    return n//2
```

   (a) What does `foo([1,2,3,4])` return?

   Answer Key: 2

   (b) What does `foo([1024,512,256,128])` return?

   Answer Key: 4

5. Given the following code:

```
file = open("numbers.txt")
total = 0
for line in file.readlines():
    for strnum in line.split(","):
        num = int(strnum)
        if num % 2 == 0:
            print(num)
            total = total + num
print(total)
```

(a) What will the output be for this `numbers.txt`?

numbers.txt:

```
5,6,7,8,9
```

Answer Key:

```
6
8
14
```

(b) What will the output be for this `numbers.txt`?

numbers.txt:

```
5
6
7
8
9
```

Answer Key:

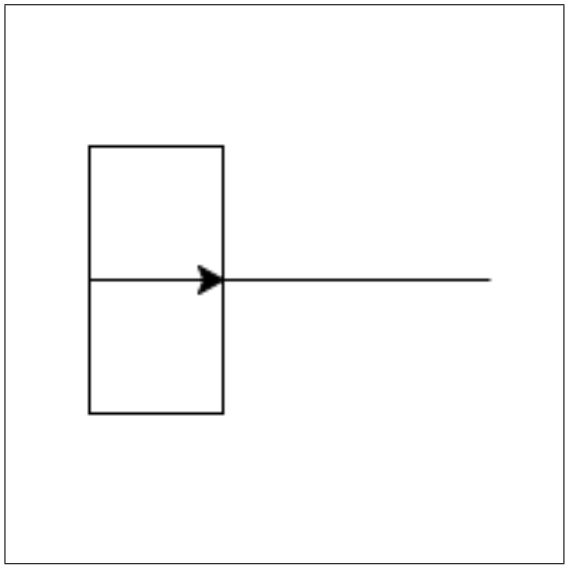
```
6
8
14
```

6. Draw what would be displayed in the graphics window when the following program is executed. Remember to indicate the final position and direction of the turtle at the end of program. (The turtle always points to the right of the screen at the start of the program.)

Graphics Displayed:

```
from turtle import *

def mystery(t, n, d):
    for i in range(n):
        if d == 'r':
            t.right(360/n)
        else:
            t.left(360/n)
        t.forward(50)

def draw(t, n):
    t.backward(100)
    mystery(t, n, 'l')
    mystery(t, n, 'r')


t = Turtle()
draw(t, 4)
```

Answer Key:



7. Write a **program** that reads in a text file, `infile.txt`, and prints out each line uppercase except for first character on each line. For example, "Hello World" should be printed out as "hELLO WORLD".

Answer Key:

```
def main():
    infile = open("infile.txt","r")
    lines = infile.readlines()
    for line in lines:
        print(line[0].lower()+line[1:-1].upper())
    infile.close()
```

8. Write the python code for the algorithms below:

(a)
```
find(st)
        set index to (length of st) - 1
        set location to -1
        set firstFound to false
        set notFound to true
        while notFound and index > -1
            if st[index] equals ',' and firstFound is false
                set firstFound to true
            otherwise, if st[index] equals ','
                set location to index
                set notFound to false
            decrement index
        return location
```

Answer Key:

```
def find(st):
    index = len(st) - 1
    location = -1
    firstFound = False
    notFound = True
    while notFound and index > -1:
        if st[index] == ',' and firstFound == False:
            firstFound = True
        elif st[index] == ',':
            location = index
            notFound = False
        index = index - 1
    return location
```

(b)
```
getSmaller(ls)
    for each item in ls
        if current item is smaller than last item in ls
            switch last item and current item in ls
```

Answer Key:

```
def getSmaller(ls):
    for i in range(len(ls)-1):
        if ls[i] < ls[-1]:
            ls[i],ls[-1] = ls[-1],ls[i]
```

9. Given the following input file `pitchers-era.dat`, write a `program` the reads in the input file and prints out the name of the pitcher followed by the word `Awesome` for each ERA between 1.00 and 1.99, the word `Great` for each ERA between 2.00 and 2.99 and the word `Good Job` for all others.

pitchers-era.dat (Name, ERA)

```
Ed Walsh, 1.82
Mariano Rivera, 2.21
Babe Ruth, 2.28
Sandy Koufax, 2.76
Juan Marichal, 2.89
Pedro Martinez, 2.93
Roger Clemens, 3.12
Greg Maddux, 3.16
```

Answer Key:

```
def main():
    infile = open("pitchers-era.dat", "r")
    for l in infile:
        words = l.split(",")
        print(words[0],end="\t")
```

```
    num = float(words[1])
    if 1.00 < num < 1.99:
        print("Awesome")
    elif 2.00 < num <  2.99:
        print("Great")
    else:
        print("Good Job")
infile.close()
```

10. Given the following code:

```
def getArray():
    input = [
        "Baggins,Frodo",
        "Gamgee,Samwise",
        "Greenleaf,Legolas",
        "Baggins,Bilbo"
        ]
    return input
def main():
    array = getArray()
    last = ""
    for line in array:
        last = line.split(",")[1]
    print(last)
main()
```

And given the following CSV file labeled `lotr.txt`:

```
Baggins,Frodo
Gamgee,Samwise
Greenleaf,Legolas
Baggins,Bilbo
```

(a) To make the program easier to update, rewrite the `getArray()` function so that it retrieves the input from the CSV file. The updated program should print the same output when run. **Answer Key:**

```
def getArray():
    infile = open("lotr.txt","r")
    input = []
    for line in infile:
        input.append(line)
    return input
```

(b) What is the output of the program?

Answer Key:

```
Bilbo
```