PROBLEM 1 : (***What are the types and values? (24 points)***)

Consider the following variables and their values for the table below.

```
words = ['tower','cake','barometer', 'cat', 'book']
say = "score 1 is 20, score 2 is 65 or 72"
```

List in the table the type of variable and its value after being assigned the expression.

variable = expression	**Type**	**Value**
`a = say[4]`	string	`"e"`
`b = say[-1]`		
`c = words[2]`		
`d = len(words)`		
`e = 5 + 4 * 2.0`		
`f = 12 % 9`		
`g = 3/7`		
`h = say.count("or")`		
`i = say.split()[:3]`		
`j = say.find("b")`		
`k = say.find("6") > say.find("2")`		
`m = words[0][-1]`		
`n = "y".join(["a","b","c"])`		

PROBLEM 2 : (***How Much? - Simple Functions (14 points)***)

A. (6 pts) Consider the problem of calculating the value of a money fund after a number of years given its current value and the compounding rate of interest that is computed annually.

Suppose `amount` is the original amount in the fund, `rate` is the rate and `years` is the number of years, then the formula for the new amount with compounded interest is:

$$newAmount = amount * (1 + rate/100)^{years}$$

Write the function `computeWithInterest` that has three integer parameters: `amount` representing the original amount, `rate` representing the interest rate, and `year` representing the number of years to compound interest. This function returns the new amount (as a decimal) calculated using the formula above.

call	**returns**	**comment**
computeWithInterest(100,5,3)	115.7625	100 * $(1 + 5/100)^3$
computeWithInterest(1000, 10, 1)	1100.0	1000 * $(1 + 10/100)^1$
computeWithInterest(1000, 6, 2)	1123.6	1000 * $(1 + 6/100)^2$

```
def computeWithInterest(amount, rate, years):
```

B. (8 pts) Rachel is ordering many frisbees for an event. Here is information on their cost.

1. Shipping costs $10 if you order less than 100, otherwise shipping is free.
2. The cost per frisbee is $2.00 each if you order less than 50.
3. The cost per frisbee is $1.50 each if you order less than 100 but more than 49.
4. The cost per frisbee is $1.00 each if you order less than 200 but more than 99.
5. The cost per frisbee is $0.80 each if you order 200 or more.

Write the function `costFrisbees` that has one integer parameter `number` representing the number of frisbees to purchase. It returns the total cost to purchase that many frisbees, including the shipping cost, if any. Examples of calls are shown.

```
def costFrisbee(number):
```

call	returns	comment
costFrisbees(5)	20.0	5 * $2 each plus $10 shipping
costFrisbees(90)	145.00	90 * $1.50 each plus $10 shipping
costFrisbees(150)	150.00	150 * $1.00 each plus no shipping cost
costFrisbees(300)	240.00	300 * $0.80 each plus no shipping cost

PROBLEM 3 : (***It's a mystery (8 points)***)

Consider the following mystery function that has two parameters, where `phrase` is a string of words and `num` is an integer. The lines have been numbered.

```
1  def mystery(phrase, num):
2      things = phrase.split()
3      answer = [ ]
4      for item in things:
5          if item.count("a") > num:
6              answer.append(item)
7      return len(answer)
```

Consider the call to mystery.

```
result =  mystery("hello giraffe aardvark cantalope fish", 1)
```

Q1. For the sample call to mystery above, what is the value assigned to result?

Q2. What type of iteration loop is the `for` loop? That is, what type is `item` and what type is `things`?

Q3. What is the first value assigned to item in the `for` loop in the sample call?

Q4. Explain in words what this function does, that is what does it calculate for any given inputs?

Q5. Suppose line 3 is changed to `answer = 0`, and line 7 is changed to `return answer`. Which additional line number needs to change for this function to compute the same value as the original code at the top of the page?

Q6. Give the new line for the previous question.

PROBLEM 4 : (***Transformations (16 points)***)

PART A (8 pts): Write the function `changeFormat` which has one string parameter `name`, where name is two or more words. In name, the last name is always one word and is the last word in name. This function returns the name in a different format, with the last name now as the first word of the name, and all words in name are now followed by a comma and a blank except for the new last word.

call	returns
changeFormat("Donald John Trump")	"Trump, Donald, John"
changeFormat("Bruce Wayne")	"Wayne, Bruce"
changeFormat("Hillary Diane Rodham Clinton")	"Clinton, Hillary, Diane, Rodham"

```
def changeFormat(name):
```

PART B (8 pts): Write the function `changeWord` which has two string parameters `word` and `letter`. This function returns the word modified with all occurrences of letter removed and all occurrences of vowels doubled, unless a vowel is the letter to remove.

For full credit, you must call the provided function `isVowel` in a useful way.

```
def isVowel(let):
    return let in "aeiouAEIOU"
```

call	returns	comment
changeWord("blackboard", "b")	"laackooaard"	all b's removed, vowels doubled
changeWord("sweetie", "e")	"swtii"	all e's removed, i doubled
changeWord("missy", "i")	"mssy"	i removed

```
def changeWord(word, letter):
```

PROBLEM 5 : (***What's Playing at the Theatre? (22 points)***)

Consider information about movies that is stored in a file in the following format. Each line represents one movie, and there are four pieces of information about that movie separated by the colon. For each movie, those four pieces of information are (in this order) the name of the movie, the genre, a movie rating (decimal number) and the year of the movie. Shown is a sample file. In the first line the name of the movie is `Zoolander 2`, the genre is `Comedy`, the rating is 2.3 and the year is 2016.

```
Zoolander 2:Comedy:2.3:2016
The Revenant:Drama:8.1:2015
The Choice:Drama:7.2:2016
How to be Single:Comedy:6.1:2016
Kung Fu Panda 3:Adventure:7.9:2016
Deadpool:Scifi:8.7:2016
Hail, Caesar!:Drama:7.1:2016
Brooklyn:Drama:7.6:2015
The Big Short:Drama:7.9:2015
The Martian:Scifi:8.1:2015
```

```
The Fifth Wave:Scifi:5.6:2016
Ride Along 2:Adventure:5.9:2016
A Walk in the Woods:Adventure:6.4:2015
The Intern:Comedy:7.2:2015
Ant-man:Scifi:7.4:2015
Burnt:Comedy:6.6:2015
Before We Go:Drama:6.9:2014
```

A function has been written named `moviedataToList` that reads in a datafile in the format above and returns a list of lists in the format shown below, where each list in the big list has four strings representing the four pieces of information about one movie from the file above. That list of lists created for the file above is partly shown below.

The line, `info = moviedataToList("moviefile.txt")`

where moviefile.txt is the file above results in

```
info = [  ['Zoolander 2', 'Comedy', '2.3', '2016'],
        ['The Revenant', 'Drama', '8.1', '2015'],
        ['The Choice', 'Drama', '7.2', '2016'],
             ... NOT ALL SHOWN
        ['Before We Go', 'Drama', '6.9', '2014']    ]
```

You will complete the method `moviedataToList` on the next page.

A. (2 pts) Consider the function `moviedataToList` which has one parameter, `datafile` that is the name of a file that is in the format on the previous page. This function reads in the file and returns a list of lists in the format also on the previous page, where each list inside the larger list is four strings representing one movie.

The function is below and has one missing line indicated by MISSING LINE.

```
def moviedataToList(datafile):
    f = open(datafile)
    answer = [ ]
    for line in f:
        line = line.strip()
        MISSING LINE
    return answer
```

What is the missing line of code?

(You could also list this code as more than one line)

B. (10 points) Write the function `numberInYear` that has two parameters named `items` and `year`, where `items` is a list of lists in the format described on the first page of this problem (each list is four strings representing one movie), and `year` is a string whose value is a year such as "1986" or "2016". This function returns the number of movies in items that are from the specified year.

Consider the two examples below. Assume info is the example list of lists on the bottom of the first page of this problem from the datafile given. Note there is 1 movie with 2014 and 8 movies with 2015 in that example file.

call	**returns**
numberInYear(info,"2014")	1
numberInYear(info, "2015")	8

```
def numberInYear(items, year):
```

C. (10 points) Write the function `highestRatingGenre` which has two parameters named `items` and `genre` where `items` is a list of lists in the format described on the first page of this problem (where each list is four strings representing one movie), and `genre` is a string whose value is genre year such as "Drama", "Comedy" or some other genre. This function returns the name of the highest rated movie with that genre. If there is more than one movie with that rating, then return the first such one found.

Consider the two examples below. Assume info is the example list of lists on the first page of this problem from the datafile given. Note that in that file, The Revenant is the highest rating of movies with genre "Drama" and The Intern has the highest rating of movies with genre "Comedy".

call	**returns**
highestRatingGenre(info,"Drama")	The Revenant
highestRatingGenre(info,"Comedy")	The Intern

```
def highestRatingGenre(items, genre):
```