1. Suppose you have a small business and are maintaining customer information in a text file: customerData.txt.

```
First Name,Middle Initial,Last Name,Street Address,City,State,Zip Code,Country,Email Address,Telephone Number,Gender,Birthday
Woodrow,C,Wilson,2362 New Street,Eugene,OR,97408,US,Woodrow.C.Wilson@spambob.com,541-337-9453,male,11/26/1984
Eric,A,Stutler,568 Nuzum Court,East Aurora,NY,14052,US,Eric.A.Stutler@trashymail.com,716-652-4943,male,11/24/1947
Rena,D,Adkins,3153 Cardinal Lane,Cleveland Heights,OH,44118,US,Rena.D.Adkins@trashymail.com,216-932-7637,female,1/14/1975
Jane,D,Smith,123 Main Street,Cedar Falls,IA,50613,US,Jane.D.Smith@gmail.com,319-555-1234,female,4/14/1970
```

In Lecture 39, we wrote a function `generateList` to read each line of text about a customer and generate a list of fields and list-of-lists of customers.

customerFields: ['First Name', 'Middle Initial', 'Last Name', 'Street Address', 'City', 'State', 'Zip Code', 'Country', 'Email Address', 'Telephone Number', 'Gender', 'Birthday']
customerList: [ ['Woodrow', 'C', 'Wilson', '2362 New Street', 'Eugene', 'OR', '97408', 'US', 'Woodrow.C.Wilson@spambob.com', '541-337-9453', 'male', '11/26/1984'],
['Eric', 'A', 'Stutler', '568 Nuzum Court', 'East Aurora', 'NY', '14052', 'US', 'Eric.A.Stutler@trashymail.com', '716-652-4943', 'male', '11/24/1947'], ... ]

Today, we want to create a mail-merge program that takes as input the customerData.txt file and a form-letter template with field names enclosed in angle-brackets (e.g., <First Name>). It should create a new directory called `formLetters` containing a customized letter for each customer. For example:

<Date>

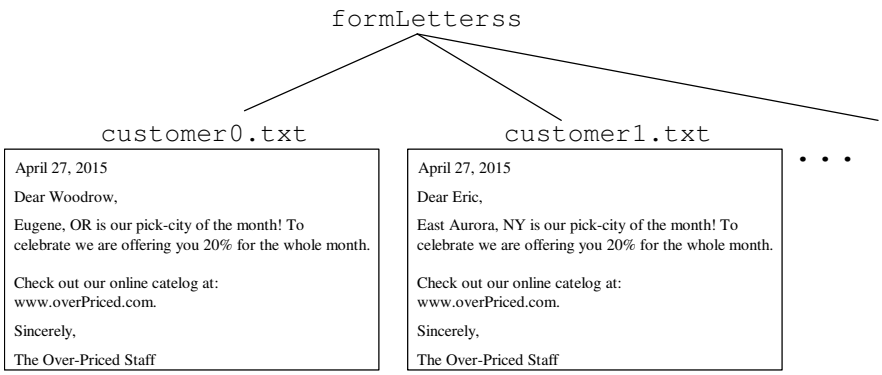
Dear <First Name>,

<City>, <State> is our pick-city of the month! To celebrate we are offering you 20% for the whole month.

Check out our online catelog at:
www.overPriced.com.

Sincerely,

The Over-Priced Staff




a) List the high-level tasks that the program must accomplish.

b) Which task(s) needs to be split into simpler sub-tasks? What are the sub-tasks?

c) Where are the loops? How would you do this by hand?

2. The `datetime` and `calendar` Python modules are useful for manipulating dates, times, and calenders. For example,

```
todayDate = date.today()
todayStr = todayDate.strftime("%B %d, %Y")
print "today",todayStr
print "year",todayDate.year
print "month",todayDate.month
print "day",todayDate.day
print "day of week", todayDate.weekday() ,
print "(Monday is 0, Tuesday is 1, etc.)"

randomDate = date(2015,5,25)
print "randomDate", randomDate.strftime("%B %d, %Y")
```

```
today April 27, 2015
year 2015
month 4
day 27
day of week 0  (Monday is 0, Tuesday is 1, etc.)
randomDate May 25, 2015
```

We can append 'Date' on the end of the customerFields and the date's string on the end of each customer's list of information.

customerFields: ['First Name', 'Middle Initial', 'Last Name', 'Street Address', 'City', 'State', 'Zip Code', 'Country', 'Email Address', 'Telephone Number', 'Gender', 'Birthday', 'Date']
customer: ['Eric', 'A', 'Stutler', '568 Nuzum Court', 'East Aurora', 'NY', '14052', 'US', 'Eric.A.Stutler@trashymail.com', '716-652-4943', 'male', '11/24/1947', 'April 27, 2015']
formLetterStr: `"\n\n<Date>\n\nDear <First Name>,\n\n<City>, <State> is our pick-city of the month! To ... -Priced Staff"`

a) Which string method(s) can locate "<" and ">" of each field tag (e.g., `<City>`) in the form letter string?

b) Which list method(s) can locate a field string (e.g., `City`) in the `customerFields` list?

c) Complete the following method which returns a completed form letter as a single string.

```
def createFormLetter(formLetterStr, customerFields, customer):
  """ Returns the form letter as a single string by replacing the customerFields
      (e.g., <First Name>) in the formLetterStr by the corresponding customer
      information. """

  startIndex =
  formLetter =
  while True:
    tagStartIndex =

    if tagStartIndex == -1:  # -l indicates not found
      formLetter +=
      break

    formLetter += formLetterStr[                    :                                ]

    tagEndIndex =

    foundTag = formLetterStr[                       :                                ]
    foundField = foundTag.strip('<>')
    fieldIndex =

    formLetter += customer[                           ]

    startIndex =

  return formLetter
```