2. This question involves two classes that are used to process log messages. A list of sample log messages is given below.

```
CLIENT3:security alert - repeated login failures Webserver:disk offline SERVER1:file not found SERVER2:read error on disk DSK1 SERVER1:write error on disk DSK2 Webserver:error on /dev/disk
```

Log messages have the format *machineld:description*, where *machineld* identifies the computer and *description* describes the event being logged. Exactly one colon (":") appears in a log message. There are no blanks either immediately before or immediately after the colon.

The following LogMessage class is used to represent a log message.

```
public class LogMessage
   private String machineId;
   private String description;
   /** Precondition: message is a valid log message. */
   public LogMessage(String message)
   { /* to be implemented in part (a) */
   /** Returns true if the description in this log message properly contains keyword;
               false otherwise.
    * /
   public boolean containsWord(String keyword)
   { /* to be implemented in part (b) */ }
   public String getMachineId()
   { return machineId; }
   public String getDescription()
   { return description; }
   // There may be instance variables, constructors, and methods that are not shown.
}
```

(a) Write the constructor for the LogMessage class. It must initialize the private data of the object so that getMachineId returns the *machineId* part of the message and getDescription returns the *description* part of the message.

Complete the LogMessage constructor below.

/\*\* Precondition: message is a valid log message. \*/
public LogMessage(String message)

Part (b) begins on page 8.

(b) Write the LogMessage method containsWord, which returns true if the description in the log message *properly contains* a given keyword and returns false otherwise.

A description *properly contains* a keyword if all three of the following conditions are true.

- o the keyword is a substring of the description;
- o the keyword is either at the beginning of the description or it is immediately preceded by a space;
- o the keyword is either at the end of the description or it is immediately followed by a space.

The following tables show several examples. The descriptions in the left table properly contain the keyword "disk". The descriptions in the right table do not properly contain the keyword "disk".

Descriptions that properly contain "disk"

"disk"		
"error	on	disk"
"error o	on	/dev/disk disk"
"error	on	disk DSK1"

Descriptions that do not properly contain "disk"

"DISK"		
"error on disk3"		
"error on /dev/disk"		
"diskette"		

WRITE YOUR SOLUTION ON THE NEXT PAGE.

Assume that the LogMessage constructor works as specified, regardless of what you wrote in part (a). Complete method containsWord below.

Part (c) begins on page 10.

(c) The SystemLog class represents a list of LogMessage objects and provides a method that removes and returns a list of all log messages (if any) that properly contain a given keyword. The messages in the returned list appear in the same order in which they originally appeared in the system log. If no message properly contains the keyword, an empty list is returned. The declaration of the SystemLog class is shown below.

```
public class SystemLog
    /** Contains all the entries in this system log.
         Guaranteed not to be null and to contain only non-null entries.
     * /
    private List<LogMessage> messageList;
    /** Removes from the system log all entries whose descriptions properly contain keyword,
         and returns a list (possibly empty) containing the removed entries.
         Postcondition:
          - Entries in the returned list properly contain keyword and
             are in the order in which they appeared in the system log.
          - The remaining entries in the system log do not properly contain keyword and
             are in their original order.
           - The returned list is empty if no messages properly contain keyword.
     * /
    public List<LogMessage> removeMessages(String keyword)
        /* to be implemented in part (c) */ }
    // There may be instance variables, constructors, and methods that are not shown.
}
```

Write the SystemLog method removeMessages, which removes from the system log all entries whose descriptions properly contain keyword and returns a list of the removed entries in their original order. For example, assume that theLog is a SystemLog object initially containing six LogMessage objects representing the following list of log messages.

```
CLIENT3:security alert - repeated login failures Webserver:disk offline SERVER1:file not found SERVER2:read error on disk DSK1 SERVER1:write error on disk DSK2 Webserver:error on /dev/disk
```

The call theLog.removeMessages("disk") would return a list containing the LogMessage objects representing the following log messages.

```
Webserver:disk offline
SERVER2:read error on disk DSK1
SERVER1:write error on disk DSK2
```

After the call, theLog would contain the following log messages.

```
CLIENT3:security alert - repeated login failures SERVER1:file not found Webserver:error on /dev/disk
```

© 2016 The College Board. Visit the College Board on the Web: www.collegeboard.org.

Assume that the LogMessage class works as specified, regardless of what you wrote in parts (a) and (b). You must use containsWord appropriately to receive full credit.

Complete method removeMessages below.

- /\*\* Removes from the system log all entries whose descriptions properly contain keyword,
- \* and returns a list (possibly empty) containing the removed entries.
- \* Postcondition:
  - Entries in the returned list properly contain keyword and
- \* are in the order in which they appeared in the system log.
- \* The remaining entries in the system log do not properly contain keyword and are in their original order.
- \* The returned list is empty if no messages properly contain keyword.

\* /

public List<LogMessage> removeMessages(String keyword)