Problem 1
Write a program that will calculate the dates of upcoming services for a seasonal subscription.

Input Variables
Season Start - the month and day of the month that the season for the subscription will start
Season End - the month and day of the month that the season for the subscription will end
Frequency - the amount of days between each service
Last Completed Service - the date of the last completed service
Amount of upcoming services to calculate - the number of services your program will calculate

Important
- If a service is within half of the frequency to the season end, the next service will be on the last day of the season.
- The season can start in one year and end in another (example: November 1st to June 1st)
- You may choose to consider all months 30 days long or by their actual amount of days
- The last completed service can be a date outside of the season.
- For the next years the first service is always in the first day of the season, for the first year it has to consider the last completed service.

Examples
Input:
Season Start - 11/1
Season End - 6/1
Frequency - 30
Last Completed Service - 2/3/2016
Amount of upcoming services to calculate - 6
Output:
3/3/16, 4/3/16, 5/3/16, 6/1/16, 11/1/16, 12/1/16

Input:
Season Start - 4/1
Season End - 8/31
Frequency - 60
Last Completed Service - 2/3/2016
Amount of upcoming services to calculate - 5
Output:
4/3/16, 6/3/16, 8/3/16, 4/1/17, 6/1/17

Input:
Season Start - 4/1
Season End - 8/31
Frequency - 30
Last Completed Service - 3/10/2016
Amount of upcoming services to calculate - 7
Output:
4/10/16, 5/10/16, 6/10/16, 7/10/16, 8/10/16, 8/31/16, 4/1/17

Problem 2

What is your understating for the query below?

Will this query have a nice performance? If not what would be a best way to solve this query?

```
SELECT
	changeLog.*,
	employees.fname,
	employees.lname
FROM
	changeLog
	JOIN employees ON (employees.employeeID = changeLog.employeeID)
WHERE
	  (changeLog.class = 8 AND changeLog.referenceID = 1397828)
	OR (changeLog.class = 4 AND changeLog.referenceID IN(
                                                        SELECT apt.appointmentID
FROM appointments apt WHERE apt.customerID = 1397828))
	OR (changeLog.class = 5 AND changeLog.referenceID IN(
                                                        SELECT tk.ticketID FROM
tickets tk WHERE tk.customerID = 1397828))
	OR (changeLog.class = 7 AND changeLog.referenceID IN(
                                                        SELECT ss.subscriptionID FROM
serviceSubscriptions ss WHERE ss.customerID = 1397828))
ORDER BY
	dateChanged DESC
```