TITLE : Lease vs. buy car: compare new car financing options to find the optimal

Description:

On one hand, buying involves higher monthly costs, but you own an asset—your vehicle—in the end. On the other hand, a lease has lower monthly payments and lets you drive a vehicle that may be more expensive than you could afford to buy, but you get into a cycle in which you never stop paying for the vehicle.

I think it is important to tackle because leasing usually costs you more than an equivalent loan because you're paying for the car during the time when it is most rapidly depreciating.

I am planning to develop a website to compare new car financing options to find the optimal in Leasing and buying a car. By developing this website, it will be easy for people to lease or buy a car. Due to severe market competition, people are opting to lease a car rather than buying a car. But, if you're looking to get a new car, you might consider leasing it instead of buying it outright. While car leases typically come with lower monthly payments, you won't actually own the car. Buying a car, on the other hand, means you'll be purchasing an asset, which can be worth making higher payments.

BUYING:

Pros of Buying a car:

- You are the owner of the vehicle you are buying
- Your car has a resale value
- You can modify and upgrade your car in any way you want.
- You own the car for as long as you wish.
- You can drive your car as much as you want as there is no minimum mileage limit while reselling the car

Cons of Buying a car:

- You have to pay the entire amount at once using cash, loan or financing which also consists of negotiated costs including cash price/down payment, taxes, insurance plus registration fee.
- You have to take care of reselling or trading your old car on your own.
- Your car will depreciate over time.

- You have to pay for the oil changes, accidental damages and maintenance of the car from your pocket.
- You have to own the car for a long term for it to be cost-efficient.

If you tend to get emotionally attached to your cars and hope to own it for the long term, buying a car is the best option. If you are taking a company car, have a business, are a professional e.g. Doctor/CA/Lawyer or need a new vehicle for a period from 2 to 4 years or a second vehicle , leasing is the best option.

For some drivers, leasing or buying is purely a matter of dollars and cents. For others, it's more about forming an emotional connection to the car. Before choosing which road to go down, it's important to understand the key distinctions. When you lease a vehicle, you're basically renting it from the dealer for a certain length of time. That's usually 36 or 48 months. Once your lease period ends, you have the option of returning the vehicle to the dealer or purchasing it at a pre-determined amount, which is defined in the lease contract. That's a lot different from buying a car. Buying it outright means you own it after the loan is paid off. Lease payments are generally lower than the monthly loan payments for a new vehicle. Monthly car loan payments are calculated based on the sale price, the interest rate, and the number of months it will take to repay the loan. The major drawback of leasing is that you don't acquire any equity in the vehicle. It's a bit like renting an apartment. You make monthly payments but have no ownership claim to the property once the lease expires. In this case, it means you can't sell the car or trade it in to reduce the cost of your next vehicle. However, there are advantages to leasing as well. They include: If you're concerned about the monthly costs, a lease eases the burden a bit. Generally, the monthly payment is considerably less than it would be for a car loan. Some people even opt for a more luxurious car than they otherwise could afford. For many people, there's nothing like the feeling of driving away in a brand new ride. If you're one of them, leasing may be the way to go. When the lease is up in a few years, you can return it and get your next new car. Many new cars offer a warranty that lasts at least three years. So when you take out a three-year lease, most of the repairs should be covered. Leasing arrangements largely eliminate the hazards of a significant unforeseen expense. With a lease, you simply return the car. The only thing you have to worry about is paying any end-of-lease fees, including those for abnormal wear or additional mileage on the vehicle.

```
import java.io.IOException;

import java.nio.file.Files;

import java.nio.file.Paths;

import java.util.List;

import java.lang.Object;
```

```
import java.util.Arrays;

import java.io.*;

import java.util.*;

import java.lang.Math;



/**

 *

 * @author sidharth

 */

public class AlgorithmsHW3 {



    /**

     *
**************************************************************************
************************

     *

     * PROBLEM SET 1

     * [10 points] fibonacci_exponential: compute nth fibonacci number
with an exponential running time

     * [10 points] fibonacci_linear: compute nth fibonacci number with
an exponential running time

     * [20 points] fibonacci_log: compute nth fibonacci number with a
logarithmic running time

     *

     * [10 points] Plot a graph showing the timings to compute the
first 30 fibonacci numbers using all three methods. And for the first
45 fibonacci numbers using the linear and logarithmic method.
```

```
                    X axis should be for the fibonacci number and y
axis should be for time.

**************************************************************************
************************

     */

    public int fibonacci_exponential(int n) {


      if(n==0){

        return 0;

     }else if(n== -1){

     return -1;

         }

         else{

        return fibonacci_exponential(n-1) + fibonacci_exponential(n-
2);

           }


    }


    public int fibonacci_linear(int n) {


      //array declaration for storing fibonacci numbers

  int a[] = new int[n + 1];

  int i;

  a[0] = 0;
```

```
  if(n > 0){
    a[1] =1;
    for(i = 2; i <= n; i++){
      a[i] = a[i - 1] + a[i - 2];
    }
  }
       return a[n];
    }
    }

    public int fibonacci_log(int n) {

     // TODO: Implement this
       return -1;
    }









    /**
     *
************************************************************************
************************
     *
```

```
     * PROBLEM SET 2

     * [20 points] You are climbing a staircase. It takes n steps to
reach the top. Each time you can either climb 1 or 2 steps. In how
many distinct ways can you climb to the top?

     * Example Input: n = 3  Output: 3 | Explanation: (1 step + 1 step
+ 1 step), (1 step + 2 steps), and (2 steps + 1 step)

     *

     * [5 points] Print out the time take to find solution for n=0 to
n=45


**************************************************************************
*************************

     */

    int climbStairs(int n) {



      // TODO: Implement this

         return -1;

    }




    /**

     *
**************************************************************************
*************************

     *

     * PROBLEM SET 3

     *
```

* [20 points] Given a triangle array, return the minimum path sum from top to bottom.

* For each step, you may move to an adjacent number of the row below (if you are on index i on the current row, you may move to either index i or index i + 1 on the next row).

* Input: triangle = [[2],[3,4],[6,5,7],[4,1,8,3]]

  Output: 11

  Explanation: The triangle looks like:

```
       2
      3 4
     6 5 7
    4 1 8 3
```

The minimum path sum from top to bottom is 2 + 3 + 5 + 1 = 11.

[5 points]

Print out the triangle (only for triangle with 4 levels) and the answer

Print out the correct answer for all triangles (from level 1 to 40)

****

*/

```
public int minimumTotal(List<List<Integer>> triangle)
{
 for (int i=0; i < triangle.size(); i++)
 {
```

```
    List<Integer> tlist = triangle.get(i);
    for (int j=0; j < tlist.size(); j++)
    {
      System.out.print(tlist.get(j)+ " ");
    }
    System.out.println();
  }
  System.out.println();



  // TODO: Implement this
     return -1;
}
```