Question 3. Prolog (weight 20%)

3a
Write a Prolog predicate `sameHead(Xs, Ys)` that is true if and only if `Xs` and `Ys` have the same (unifiable) first elements

E.g.

```
| ?- sameHead([1,2,3],[1,4,5]).  
  yes
| ?- sameHead([X,1],[2,3]).     
  X = 2
| ?- sameHead([1,2],[3,4]).     
  no
```

3b
Write a Prolog predicate `zip(Xs, Ys, Zs)` that is true if and only if `Xs` and `Ys` are lists of the same length, and `Zs` consists of the first element of `Xs`, followed by the first element of `Ys`, then the second of `Xs`, the second of `Ys`, etc.

E.g.

```
| ?- zip([1,2,3],[a,b,c],[1,a,2,b,3,c]).  
  yes
| ?- zip([1],[a,b],[1,a,b]).       
  no
```

3c
Use `zip` to define a Prolog predicate `stutter(Xs)` that is true if and only if `Xs` is a list with an even number of elements, where the first element is equal to (i.e. is unified with) the second, the third is equal to (unified with) the fourth, etc. E.g.

```
| ?- stutter([a,a,b,b]).   
  yes
| ?- stutter([1,1,1,1,1,1]). 
  yes
| ?- stutter([1,a,a,1]).   
  no
```
3d

What will the Prolog interpreter answer to the following queries:

\[ \text{?- zip([X],[Y],[a,b])}. \]
\[ \text{?- zip([X],[Y],[Y,X])}. \]
\[ \text{?- stutter([X,a])}. \]
\[ \text{?- stutter([X,Y,Z,a,z,Y,X,b])}. \]

3e

Briefly explain:
- what the effect of the cut predicate is
- why it is needed in Prolog
- why it is problematic
- what the effect of the cut predicate is
- why it is needed in Prolog
- why it is problematic