1. **(3 points) Drop every N'th element from a list.**
   
   Example:
   ```prolog
   ?- drop([a,b,c,d,e,f,g,h,i,k],3,X).
   X = [a,b,d,e,g,h,k]
   ```
   
   Hint: The arithmetic operators are: \(=\),\(<\),\(<=\),\(=:=\),\(/\),\(*\),\(+\),\(-\),\(\text{mod}\),\(\text{div}\)
   
   To perform arithmetic, use "is" not "=".
   
   I found it useful to create `split(L,N,L1,L2)` which took list `L` and split it into two lists, the first of which is `N` in length.
   
   Hint: typing "a" aborts the execution.

2. **(3 points) Extract a slice from a list.**
   
   Given two indices, I and K, the slice is the list containing the elements between the I'th and K'th element of the original list (both limits included). Start counting the list elements with 1.
   
   Example:
   ```prolog
   ?- slice([a,b,c,d,e,f,g,h,i,k],3,7,L).
   X = [c,d,e,f,g]
   ```

3. **(3 points) Rotate a list N places to the left.**
   
   Examples:
   ```prolog
   ?- rotate([a,b,c,d,e,f,g,h],3,X).
   X = [d,e,f,g,h,a,b,c]
   ```
   ```prolog
   ?- rotate([a,b,c,d,e,f,g,h],-2,X).
   X = [g,h,a,b,c,d,e,f]
   ```
   
   Hint: Use the predefined predicates `length/2` and `append/3`, as well as the results of problems you have written.

4. **(3 points) Create a list containing all integers within a given range.**
   
   Example:
   ```prolog
   ?- range(4,9,L).
   L = [4,5,6,7,8,9]
   ```
5. **(3 points) Extract a given number of randomly selected elements from a list.**
The selected items shall be put into a result list.
Example:
?- ranList([a,b,c,d,e,f,g,h],3,L).
L = [e,d,a]

Hint: Use the built-in random number generator random/3.
random(L, U, R) which binds R to a random number in [L,U].

Hint: I used rotate.

6. **(10 points) Use prolog to solve the logic puzzle below:**

Logan's local library recently sponsored a writing contest for young children in the community. Each of the four contestants (including Ralph) took on the task of bringing to life an imaginary friend in a short story. Each child selected a different type of animal (including a moose) to personify, and each described a different adventure involving this new friend (one story described how an imaginary friend had formed a rock band). From the following clues, can you match each young author with his or her imaginary friend and determine the adventure the two had together?
   a) The seal (who isn’t a creation of either Joanne or Lou) neither rode to the moon nor took a trip on a magic train.
   b) Joanne’s imaginary friend (who isn’t a grizzly bear) went to the circus.
   c) Winnie’s imaginary friend is a unicorn.
   d) The grizzly bear didn’t go to the moon.

Hint: As an example, the solution to the puzzle below is on the class website. Try it on your own, before peeking.

1. Four couples in all attended a costume ball.
2. The lady dressed as a cat arrived with her husband Matt.
3. Two couples were already there, one man dressed like a bear.
4. First to arrive wasn't Vince, but he got there before the Prince.
5. The witch (not Sue) is married to Chuck, who was dressed as Donald Duck.
6. Mary came in after Lou, both were there before Sue.
7. The Gipsy arrived before Ann, neither is partner to Batman.
8. If Snow White arrived after Tess, then how was each couple dressed?

Specifics:
The following lists of possible choices may be helpful.
man([vince,chuck,lou,matt]).
woman([sue,mary,ann,tess]).
costumeMan([batman, donald_duck, prince, bear]).
costumeWoman([witch, gipsy, cat, snow_white])

I used a list of lists as my data structure. So a solution might look like (where the order is the order in which they arrived):
I used equalSet to make sure that the set of answers I got (for any category) was exactly equal to the set of possible answers (so no one was left out and every name was used only once).

There are several forms of equal (and not equal). **Note:** \( Arg1 \equiv Arg2 \) succeeds if \( Arg1 \) and \( Arg2 \) are not literally identical.

Use "trace" to see where your problems are.
You can use an "or" by placing a ";" between parenthesized goals.