

Logic Programming

General Exercises (*Bonus, not mandatory*)

1. (2 Points) Write a prolog program to find whether the length of a list is even or odd.
2. (2 Points) Write a prolog program to sum odd numbers between given range.
3. (2 Points) Write a prolog program to add an element to the last position in given list.
4. (2 Points) Consider a representation of sets as lists. Define the following predicates:
 - a. member(X,L), which holds if the element X occurs in L.
 - b. subset(L,K), which holds if L is a subset of K.
 - c. disjoint(L,K), which holds if L and K are disjoint (i.e. they have no elements in common).
 - d. union(L,K,M), which holds if M is the union of L and K.
 - e. intersection(L,K,M), which holds if M is the intersection of L and K.
 - f. difference(L,K,M), which holds iff M is the difference of L and K.
5. (2 Points) Define a predicate occurrences(X,L,N) which holds if the element X occurs N times in the list L.
6. (2 Points) Write a prolog predicate pretty_print(L) to print the elements of a given list (L) as following:


```
?- pretty_print( [2, 3, [4,5,[1,2]], 9])
2
3
    4
    5
        1
        2
9
```
7. (2 Points) Write a prolog program to prompt the user for two numbers and return a list of consequence integer numbers starts from the smallest and ends with the largest given number.


```
? - Enter two numbers:
:| 3.
:| 7.
[3, 4, 5, 6, 7]
```
8. (2 Points) Write a prolog predicate classify(L) to classify the elements of a given list (L), whether each element is Number, Atom, Character or List.

?- classify([6, orange, 'a', [2,3]])

6 is number

orange is atom

'a' is character

[2,3] is list

- 9. (2 Points) Write a prolog program to continually read an Integer number and print it's square, then to ask the user whether to continue or not. If the user enter Y, the program will repeat again, if N the program will terminate.**
- 10. (2 Points) Write a prolog predicate count(File, Character, Occurrence) to count the occurrence of a given character in a given file .**