

Assignment 4

Due: Friday, 02.06.2017, 15:59 via Git

For help, contact alp-staff@lists.iai.uni-bonn.de (staff only) or
alp-course@lists.iai.uni-bonn.de (staff and participants).

Start working on the exercises early enough so that you can contact your tutor in time if you have problems. Don't expect your tutor to be available at midnight or during weekends!

Submit results into the folder "assignment04/" of the git repository of your group.

For each task, submit your implemented predicate as a file named "taskN.pl". At the bottom of the file add a comment containing console output that shows **all results of a successful test run of your predicate**.

Task 1. Application of substitution (3 Points)

Write down the term resulting from applying the respective substitution, or explain why the substitution cannot be applied. For example, write " $f(X)\{X\leftarrow 1\} \equiv f(1)$ ".

- a) $f(X,Y)\{X\leftarrow 'Z'\}$ \equiv
- b) $g(X,Y)\{X\leftarrow 2,Y\leftarrow g(X)\}$ \equiv
- c) $h(X,Y)\{X\leftarrow h(Z,Y),Y\leftarrow h(Z),Z\leftarrow 3\}$ \equiv

Task 2. Linear list (11,5 points)

Implement a predicate "linear(+NestedList, ?LinearList)" that succeeds whenever LinearList is a list whose elements may themselves be arbitrarily deeply nested lists and LinearList contains all elements of NestedList in the same order but without any nesting. For instance,

?-linear ([1,[2,3,[a,[b],c]]], [1,2,3,a,b,c]).

should succeed but

?-linear ([1,[2,3,[a,[b],c]]], [1,2,3,a,c,b]).

should fail.

Tip: You may use the predefined predicate `append(A, B, AB)` to implement your version of `linear/2`. Except for that, your solution must be self-contained.

Task 3. *Grouping consecutive list elements (11 Points)*

Write a predicate that groups consecutive repeated elements of a list into sublists. If a list contains non-consecutive repeated elements they should be placed in separate sublists.

Example:

```
?- group([1,1,1,1,2,c,c,1,1,d,e,e,e],X).
```

```
X = [[1,1,1,1],[2],[c,c],[1,1],[d],[e,e,e,e]]
```

Task 4. *Grouping consecutive list elements (4,5 Points)*

Modify your predicate from Task 4 so that it does not put an element into a sublist if there is no consecutive repeated element. Example:

```
?- group([1,1,1,1,2,c,c,1,1,d,e,e,e],X).
```

```
X = [[1,1,1,1],2,[c,c],[1,1],d,[e,e,e,e]]
```

Tip: In both cases (Task 4 and 5) the essential question is “How can your predicate remember whether it had seen the same element in the previous step?”

Task 5. *Slice of a list (11 Points)*

Given a list L and two indices, I and J, the slice of L from I to J is the list containing the elements of L from position I to position J (both limits included). Start counting the elements with 1.

Example:

```
?- L=[a,b,c,d,e,f,g,h,i,k], I=3, J=7, slice(L, I,J,Slice).
```

```
Slice = [c,d,e,f,g]
```