

Structure of Programming Languages – Lecture 5

CS 636 – 536

May 4, 2009

- 1 Structured Control
 - Loops

- 2 Other Iteration
 - Prolog

Loops

- `while` : test at the top.
- `do ... while`: test at the bottom.
- `begin ... again`: explicit infinite loop, usually used with `if ... break`
- counted loops : fixed trip-count or variable
- for-all loops

Prolog: Database with recursive backtracking search

Conditional clauses (rules)	<pre>pretty(X) :- artwork(X). pretty(X) :- color(X,red), flower(X). watchout(X) :- sharp(X,_).</pre>
-----------------------------	--------------------------------------------------------------------------------------------------------------

Unconditional clauses (facts)	<pre>color(rose, red). sharp(rose, stem). sharp(holly, leaf). flower(rose). flower(violet). artwork(painting(Monet, haystack_at_Giverny))</pre>
-------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------

- 1 Work on the leftmost subgoal first.
- 2 Select the first applicable rule.
- 3 Search the facts in the order they appear in the data base.

Prolog queries.

```
?- pretty(rose).
```

```
yes
```

```
?- pretty(Y).
```

```
Y=painting(Monet, haystack_at_Giverny).
```

```
Y=rose
```

```
no
```

```
?- pretty(W),sharp(W,Z).
```

```
W=rose Z=stem
```

```
no
```

Prolog computation.

```
gcd(A, 0, A).  
gcd(A, B, D) :- (A>B), (B>0),  
    R is A mod B,  
    gcd(B, R, D).  
gcd(A, B, D) :- (A<B), gcd(B, A, D).
```

```
factorial(0,1).  
factorial(N,F) :-  
    N>0,  
    N1 is N-1,  
    factorial(N1,F1),  
    F is N * F1.
```