

Mengenprädikate

Übersicht

- ◆ Lösungsmengen als Daten
- ◆ Alle Lösungen für ein Ziel erhalten (*all solutions*)
 - ◆ findall/3 – Ohne Backtracking und Variablenbindung
 - ◆ Mit Backtracking auf ungebundene Variablen
 - ◆ bagof/3 – Lösungsliste kann Duplikate enthalten
 - ◆ setof/3 – Lösungsliste ist Menge
 - ◆ Explizites Binden von Variablen durch Existenzquantor \exists
- ◆ Anwendungen von Mengenprädikaten

Mengenprädikate – 1

findall/3 – Lösungslisten finden

```
word(cat, n).  
word(ring, n).  
word(do, v).  
word(ring, v).
```

```
?- findall(Word, word(Word, Cat), Words).  
Words = [cat,ring,do,ring] ? ;  
no
```

Finde alle Wörter, egal von welcher Kategorie.

Das eingebaute Prädikat findall/3

- ◆ berechnet alle Lösungen eines Ziels.
- ◆ liefert gewünschte Teile daraus als Elemente einer Liste zurück.
 - ▶ Die Reihenfolge der Elemente entspricht der Reihenfolge, in der die Lösungen gefunden werden.
- ◆ ist wichtig, wenn alle Lösungen als Ganzes weiter verarbeitet werden sollen.

Mengenprädikate – 2

findall/3

```
?- findall(Word/Cat,  
          (word(Word,Cat), word(Word,Cat2), Cat \== Cat2),  
          Ambig).  
Ambig = [ring/n,ring/v] ? ;  
no
```

Finde alle kategoriell ambigen Wörter mit ihren möglichen Kategorien.

findall(Term, Ziel, Liste)

- ◆ *Term* – wird für jede Lösung von *Ziel* zu *Liste* hinzugefügt
- ◆ *Ziel* – Ziel, das zu beweisen ist
- ◆ *Liste* – enthält für jede Lösung von *Ziel* die entsprechende **Instanz** von *Term*

Mengenprädikate – 3

findall/3 – Kontrollverhalten

```
?- findall(X, fail, Resultat).  
Resultat = [] ? ;  
no
```

Kontrollverhalten

- ◆ falls das *Ziel* einfach fehlschlägt, ist die *Liste* leer
- ◆ freie Variablen in *Term* und *Ziel* werden **nie** gebunden!
- ◆ findall/3 terminiert nur, wenn der Suchbaum von *Ziel* endlich ist
- ◆ findall/3 gelingt höchstens einmal
- ◆ findall/3 kann nur scheitern, wenn die *Liste* instantiiert aufgerufen wird!

```
?- findall(Cat, word(Word, Cat), [n,v,a]).  
no
```

Mengenprädikate – 4

bagof/3

```
word(cat, n).
word(ring, n).
word(do, v).
word(ring, v).

?- bagof(Word, word(Word,Cat), Words).
Cat = n, Words = [cat,ring] ? ;
Cat = v, Words = [do,ring] ? ;
no
```

Finde für jede Kategorie alle Wörter.

Das eingebaute Prädikat `bagof(Term, Ziel, Liste)` funktioniert wie `findall/3`, aber

- alle freien Variablen in *Ziel*, die nicht in *Term* vorkommen, werden gebunden, und *Liste* jeweils für eine unterschiedliche Bindung berechnet.
- falls *Ziel* nicht erfüllt werden kann, scheitert `bagof/3`.

Mengenprädikate – 5

Existenzquantor $\wedge/2$

```
word(cat, n).
word(ring, n).
word(do, v).
word(ring, v).

?- bagof(Word, Cat^word(Word,Cat), Words).
Words = [cat,ring,do,ring] ? ;
no
```

Finde alle Wörter, von welcher Kategorie auch immer.

Der Existenzquantor \wedge bindet freie Variablen in *Ziel*.

- $\wedge/2$ erlaubt es, die Lösungen für alle möglichen Belegungen für die Variable auf der linken Seite des Operators zu berechnen.
- $\wedge/2$ ist ein rechts-assoziativer Operator

```
?- bagof(W, C1^C2^(word(W,C1),word(W,C2),C1 \== C2), W).
```

Mengenprädikate – 6

setof/3 – sortierte Lösungsmenge

```
word(cat, n).
word(ring, n).
word(do, v).
word(ring, v).
word(nice, a).

?- setof(Cat, Word^word(Word,Cat), Cats).
Cats = [a,n,v] ? ;
no
```

Finde die Menge aller Kategorien von welchen Wörtern auch immer.

Das eingebaute Prädikat `setof(Term, Ziel, Liste)` funktioniert wie `bagof/3`, aber

- die Liste enthält keine Duplikate.
- die Liste ist entsprechend der Standardordnung für Terme sortiert.

```
setof(Term, Ziel, Menge) :-
    bagof(Term, Ziel, Liste),
    sort(Liste, Menge).
```

Mögliche Definition von `setof/3`

Mengenprädikate – 7

Anwendungen von Mengenprädikaten

Mengenprädikate

- sprengen den Rahmen der Prädikatenlogik erster Stufe.
- erlauben Dinge zu berechnen, die uns bis anhin unmöglich waren.
 - Wie viele Lösungen hat eine Anfrage?
 - Verschiedene Lösungen einer Anfrage vergleichen.
- Beispiel: Welcher Anteil von Wörtern im Lexikon ist ambig?

```
ambig(Anteil) :-
    setof(W, Cat^word(W,Cat), Ws),
    length(Ws, AnzahlWoerter),
    setof(A, C1^C2(word(A,C1),word(A,C2),C1 \== C2), As),
    length(As, AnzahlAmbige),
    Anteil is AnzahlAmbige / AnzahlWoerter.
```

Mengenprädikate – 8