# Лекция 9 (13.04.2022)

## Алгоритм унификации

Алгоритм унификации — это основной шаг доказательства. Предназначен для того, чтобы подобрать знания *для того, чтобы ответить на очередной вопрос* (чтобы проверить можно ли унифицировать два терма?). Алгоритм унификации — это процесс => использует какие-то области памяти. Алгоритм унификации использует рабочее поле, стек (для хранения равенств, возможность унификации которых проверяется), результирующая ячейка (содержит подстановку – множество пар Xi = Ti) и флаг.

Начало  
 Занести T1 = T2  
 Положить Неудача = 0  
 Пока стек не пуст  
 Считать из стека в раб. Поле S = T  
 Обработать  
 а) Если S и T несовпадающие константы, то Неудача = 1, выход из цикла.  
 б) Если S и T совпадающие константы, след. шаг цикла (на конец цикла).  
 в) Если S – переменная, а T – терм содержащий S, то Неудача = 1 и выход из цикла.  
 г) Если S – переменная, а T – терм не содержащий S, то Отыскать в стеке и в результирующей ячейке все вхождения S и заменить их на T. Добавить в результирующую ячейку S = T.

д) Если S и T составные термы, имеющие разные функторы или разную R-ность, то Неудача = 1 и выход из цикла.

е) Если S и T составные термы с одинаковыми функторами (F) и R-ностью f(S1, S2, … Sk), f(T1, T2, … Tk), то занести в стек S1 = T1, S2 = T2, … Sk = Tk.  
 Очистить раб. Поле  
 Конец цикла  
 Если в результате Неудача = 1, то унификация невозможна, если Неудача = 0, то унификация успешна, а результирующая ячейка содержит подстановку – наиболее общий унификатор.

Терм S называется более общим чем T, если S явл. примером T, а T не явл. примером S

Терм S называется более общим примером двух термов Т1 и Т2, если S такой их общий пример, который является наиболее общим по отношению к любому другому их примеру.

Унификатором двух термов называется подстановка, которая, будучи применена к двум термам даст одинаковый результат.

Наиболее общим унификатором двух термов называется унификатор соотв. примеру двух термов.

Теорема – если два терма унифицируемы, то существует единственный, с точностью до переименования переменных, наиболее общий унификатора (а значит наиболее общий пример).

Поэтому алгоритм унификации и строит этот наиболее общий пример.

t(X, p(X, Y)) = t(q(W), p(q(a), b)))

|  |  |  |  |
| --- | --- | --- | --- |
| Ш | Рез. ячейка | Раб. поле | Стек |
| 0 |  |  | t(X, p(X, Y)) = t(q(W), p(q(a), b))) |
| 1 |  | t(X, p(X, Y)) = t(q(W), p(q(a), b))) | X = q(W), p(X, Y) = p(q(a), b) |
| 2 | X = q(W) | X=q(W) | p(X, Y) = p(q(a), b) |
| 3 | X = q(W) | p(q(W), Y) = p(q(a), b) | q(W) = q(a), Y = b |
| 4 | X = q(W) | q(W) = q(a) | W = a, Y = b |
| 5 | X = q(W) W = a | W = a | Y = b |
| 6 | X = q(W)  W = a Y = b | Y = b |  |

Итого:

## Общая схема согласования целевых утверждения

Решение задачи с помощью логической программы начинается с вопрос G и завершается получением одного из двух результатов:

1. Успешного согласование вопроса и программы (базы знаний), а в качестве побочного эффекта будет получена подстановка, которая содержит значения переменных, при которых вопрос является примером программы, или несколько таких примеров.
2. Неудача.

Вычисления с помощью конечной логической программы представляют собой пошаговое преобразования исходного вопроса. На каждом шаге имеется некоторая конъюнкция целей выводимость, которой следует доказать. Эта конъюнкция целей называется резольвентой (Резольвента хранит цели, которые надо доказать. Это динамическая структура, хранится в виде стека). Успех достигается тогда, когда резольвента пуста.

Преобразование резольвенты выполняется с помощью операции, которая называется редукция.

Редукция цели G с помощью программы P, называется замена цели G телом того правила из P, заголовок которого унифицируется с целью. (Такие правила обычно называется сопоставимыми с целью)

Новая резольвента образуется в 2 этапа:

1. В текущей резольвенте, выбирается одна из целей, и для неё выполняется редукция.
2. Затем, к полученной конъюнкции целей, применяется подстановка, полученной как наибольший общий унификатор цели и заголовка сопоставленного с ней правила.

Если для редукции был выбран факт, то резольвента уменьшится на одну цель.

Если применяется правило, то число целей в резольвенте не уменьшается.

Процесс прекратится когда

1. Будет получен пустая резольвента => успех доказательства
2. Очередную подцель невозможно унифицировать ни с одним дизъюнктом программы(неудача) => тогда откат (механизм возврата) к той точке, в которой выбирался унифицирующийся с последней подцелью дизъюнкт. Для этого точка, где выбирался один из возможных унифицируемых с подцелью дизъюнктов, запоминается в специальном стеке, для последующего возврата к ней и выбора альтернативы в случае неудачи.