|  |  |  |  |
| --- | --- | --- | --- |
| No шага | Состояние резольвенты, и вывод: дальнейшие действия (почему?) | Для каких термов запускается алгоритм унификации: Т1=Т2 и каков результат (и подстановка) | Дальнейшие действия: прямой ход или откат (почему и к чему приводит?) |
| 1 | fact(3, Res) | rec\_fact(N, Res, Acc) и fact(3, Res)  **Результат:** неудача, термы не унифицируемы | Прямой ход. Переход к следующему предложению. |
| 2 | fact(3, Res) | rec\_fact(\_, Res, Acc) и fact(3, Res)  **Результат:** неудача, термы не унифицируемы | Прямой ход. Переход к следующему предложению. |
| 3 | fact(3, Res), ! | fact(N, Res) и fact(3, Res)  **Результат:** успех + подстановка {N = 3, Res = Res} | Заголовок правила заменяется его телом c учетом подстановки:  rec\_fact(3, Res, 1) |
| 4 | rec\_fact(3, Res, 1) | rec\_fact(N, Res, Acc) и rec\_fact(3, Res, 1)  **Результат:** успех + подстановка {N = 3, Res = Res, Acc = 1} | Заголовок правила заменяется его телом c учетом подстановки:  3 > 1, NewN = 3 - 1, NewAcc = 1 \* 3, rec\_fact(NewN, Res, NewAcc), ! |
| 5 | 3 > 1, NewN = 3 - 1, NewAcc = 1 \* 3, rec\_fact(NewN, Res, NewAcc), ! | 3 > 1  **Результат:** успех | Переход к следующему терму |
| 6 | NewN = 3 - 1,  NewAcc = 1 \* 3, rec\_fact(NewN, Res, NewAcc), ! | NewN = 3 – 1  **Результат:** успех + {NewN = 2} | Переход к следующему терму c учётом подстановки NewN = 2 |
| 7 | NewAcc = 1 \* 3, rec\_fact(2, Res, NewAcc), ! | NewAcc = 1 \* 3  **Результат:** успех | Переход к следующему терму c учётом подстановки NewAcc = 3 |
| 8 | rec\_fact(2, Res, 3), ! | rec\_fact(2, Res, 3) и rec\_fact(N, Res, Acc)  **Результат:** успех + подстановка {N = 2, Res = Res, Acc = 3} | Заголовок правила заменяется его телом c учетом подстановки:  2 > 1, NewN = 2 - 1, NewAcc = 3 \* 2, rec\_fact(NewN, Res, NewAcc), ! |
| 9 | 2 > 1, NewN = 2 - 1, NewAcc = 3 \* 2, rec\_fact(NewN, Res, NewAcc), !, ! | 2 > 1  **Результат:** успех | Переход к следующему терму |
| 6 | NewN = 2 - 1,  NewAcc = 3 \* 2, rec\_fact(NewN, Res, NewAcc), !, ! | NewN = 2 – 1  **Результат:** успех + {NewN = 1} | Переход к следующему терму c учётом подстановки NewN = 1 |
| 7 | NewAcc = 3 \* 2, rec\_fact(1, Res, NewAcc), !, ! | NewAcc = 3 \* 2  **Результат:** успех | Переход к следующему терму c учётом подстановки NewAcc = 6 |
| 8 | rec\_fact(1, Res, 6), !, ! | rec\_fact(1, Res, 6) и rec\_fact(N, Res, Acc)  **Результат:** успех + подстановка {N = 1, Res = Res, Acc = 6} | Заголовок правила заменяется его телом c учетом подстановки:  1 > 1, NewN = 1 - 1, NewAcc = 6 \* 1, rec\_fact(NewN, Res, NewAcc), ! |
| 9 | 1 > 1, NewN = 1 - 1, NewAcc = 6 \* 1, rec\_fact(NewN, Res, NewAcc), !, !, ! | 1 > 1  **Результат:** неудача | **Откат к 8** |
| 10 | rec\_fact(1, Res, 6), !, ! | rec\_fact(\_, Res, Acc) и rec\_fact(1, Res, 6)  **Результат:** успех + подстановка {\_ = 1, Res = Res, Acc = 6} | Заголовок правила заменяется его телом c учетом подстановки:  Res = 6 |
| 11 | Res = 6, !, ! | Res = 6  **Результат:** успех | Переход к следующему терму |
| 12 | !, ! | !  **Результат:** успех | Переход к следующему терму |
| 13 | ! | !  **Результат:** успех | Переход к следующему терму |
| 14 |  |  | Найдено решение Res=6  Отсечение => система больше не будет искать решение в процедуре rec\_fact  Откат к 3 |