

Но шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: T1=T2 и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (почему и к чему приводит?)
1	length(Res, [1, 2])	length_tmp(Res, Len, [_   Tail]) и length(Res, [1, 2]) <b>Результат:</b> неудача, термы не унифицируемы	Прямой ход. Переход к следующему предложению.
2	/--/	/--/	/--/
3	length(Res, [1, 2])	length(Res, [1, 2]) и length(Res, List) <b>Результат:</b> успех + подстановка {List = [1, 2], Res = Res}	Заголовок правила заменяется его телом с учетом подстановки
4	length_tmp(Res, 0, [1, 2]), !	length_tmp(Res, Len, [_   Tail]) и length_tmp(Res, 0, [1, 2]), <b>Результат:</b> успех + подстановка {Len = 0, Res = Res, Tail = [2]}	Заголовок правила заменяется его телом с учетом подстановки
5	NewLen = 0 + 1, length_tmp(Res, NewLen, [2]), !	NewLen = 0 + 1 <b>Результат:</b> успех	Переход к следующему терму
6	length_tmp(Res, 1, [2]), !	length_tmp(Res, 1, [2]) и length_tmp(Res, Len, [_   Tail]) <b>Результат:</b> успех + подстановка {Len = 1, Res = Res, Tail = []}	Заголовок правила заменяется его телом с учетом подстановки:
7	NewLen = 1 + 1, length_tmp(Res, NewLen, []), !	NewLen = 1 + 1 <b>Результат:</b> успех	Переход к следующему терму
8	length_tmp(Res, 2, []), !	length_tmp(Res, Len, [_   Tail]) и length_tmp(Res, 2, []) <b>Результат:</b> неудача, термы не унифицируемы	Прямой ход. Переход к следующему предложению.
9	length_tmp(Res, 2, []), !	length_tmp(Res, 2, []) и length_tmp(Res, Len, []) <b>Результат:</b> успех + подстановка {Len = 2, Res = Res}	Заголовок правила заменяется его телом с учетом подстановки
10	Res = 2, !, !	Res = 2 <b>Результат:</b> успех	Переход к следующему терму
11			Найдено решение Res=2 Отсечение => система больше не будет искать решение в процедуре length_tmp