

# Логическое программирование

Кевролетин В.В. группа с8403а(246)

19 May 2012

## Содержание

<b>1</b>	<b>Задание</b>	<b>1</b>
1.1	Условие . . . . .	1
1.2	Решение . . . . .	1
1.2.1	Исходный код . . . . .	1
1.2.2	Тесты . . . . .	2

## 1 Задание

### 1.1 Условие

(3.3iii) Написать программу для отношения `no_doubles/2`.

### 1.2 Решение

#### 1.2.1 Исходный код

`member_(Xs, Y)` истина, если `Y` содержится в списке `Xs`

`Xs` список

`Y` произвольный тип

`not_member(Xs, Y)` истина, если `Y` не содержится в списке `Xs`

`Xs` список

`Y` произвольный тип

```
member_([], _) :- false.
member_([X | _], X).
member_([_ | Xs], Y) :-
    list(Xs),
    member_(Xs, Y).
```

```
not_member([], _).
not_member([X | _], X) :- false.
not_member([X | Xs], Y) :-
    list(Xs),
```

```
\=(X, Y),
not_member(Xs, Y).
```

**no\_doubles(Xs, Ys)** истина, если Ys получается из Xs после удаления повторяющихся элементов (для каждой группы повторяющихся элементов оставляется последний)

**Xs** Список

**Ys** Список

```
no_doubles([], []).
no_doubles([X | Xs], [X | Ys]) :-
    list(Xs),
    not_member(Xs, X),
    no_doubles(Xs, Ys).
no_doubles([X | Xs], Ys) :-
    list(Xs),
    member_(Xs, X),
    no_doubles(Xs, Ys).
```

### 1.2.2 Тесты

- no\_doubles(+, +)

```
?- no_doubles([1, 2, 3], [1, 2, 3]).
```

**true**

```
?- no_doubles([1, 1, 2, 3, 1], [2, 3, 1]).
```

**true**

```
?- no_doubles([1, 1, 2, 3, 1], [1, 2, 3, 1]).
```

**false.**

- no\_doubles(+, -)

```
no_doubles([a, a, a, b, b, c, b, d, e, e, a], X).
```

```
X = [c,b,d,e,a] ?
```

```
no_doubles([], X).
```

```
X = []
```

```
no_doubles([1, 2, 3], X).
```

```
X = [1, 2, 3]
```

?- no\_doubles([1, 2, 3], X).

X = [1, 2, 3]

?- no\_doubles([1, 2, 3, 2, 3, 1], X).

X = [2, 3, 1]

?- no\_doubles([1, 2, 3, 4, 2, 3, 1], X).

X = [4, 2, 3, 1]