

# Логическое программирование

Кевролетин В.В. группа с8503а(256)

12 November 2012

## Содержание

<b>1</b>	<b>Задание 6</b>	<b>1</b>
1.1	Условие . . . . .	1
1.2	Решение . . . . .	1
1.2.1	Исходный код . . . . .	1
1.2.2	Тесты . . . . .	3

## 1 Задание 6

### 1.1 Условие

Придумать формализацию с другим ходом типа `fill_empty_transfer` для задачи о сосудах (8,5,4).

### 1.2 Решение

Рассмотрим решение, полученное программой до введения нового хода:

```
[ fill(1), fill(2), empty(1), transfer(2, 1),  
  fill(2), transfer(2, 1), empty(1), transfer(2, 1),  
  fill(2), transfer(2, 1), fill(2), transfer(2, 1) ]
```

В приведенном решении несколько раз заполняется меньший сосуд, его содержимое выливается в больший сосуд и полученный остаток используется далее. Три последовательных хода могут быть объединены в один составной ход `fill_empty_transfer`

```
update(jugs(V1,V2),fill_and_transfer(1), jugs(0,V)) :-  
    capacity(1, C1),  
    capacity(2, C2),  
    >(C1, C2),  
    V is (C1 + V2) mod C2.
```

#### 1.2.1 Исходный код

Полный текст программы:

- Фреймворк для решения задач поиском в глубину

```

solve_dfs(State, _, []) :-
    final_state(State).
solve_dfs(State, History, [Move|Moves]) :-
    move(State, Move),
    update(State, Move, State1),
    legal(State1),
    \+ member(State1, History),
    solve_dfs(State1, [State1|History], Moves).

test_dfs(Moves) :-
    initial_state(State), solve_dfs(State, [State], Moves).

```

- Задача о сосудах

```

initial_state(jugs(0,0)).

final_state(jugs(4,_)).
final_state(jugs(_,4)).

move(jugs(_,_), fill_and_transfer(1)).
move(jugs(_,_), fill(1)).
move(jugs(_,_), fill(2)).
move(jugs(_,_), empty(1)).
move(jugs(_,_), empty(2)).
move(jugs(_,_), transfer(2,1)).
move(jugs(_,_), transfer(1,2)).

update(jugs(V1,V2), fill_and_transfer(1), jugs(0,V)) :-
    capacity(1, C1),
    capacity(2, C2),
    >(C1, C2),
    V is (C1 + V2) mod C2.
update(jugs(_,V2), fill(1), jugs(C1,V2)) :- capacity(1,C1).
update(jugs(V1,_), fill(2), jugs(V1,C2)) :- capacity(2,C2).
update(jugs(_,V2), empty(1), jugs(0,V2)).
update(jugs(V1,_), empty(2), jugs(V1,0)).
update(jugs(V1,V2), transfer(2,1), jugs(W1,W2)) :-
    capacity(1,C1),
    Liquid is V1 + V2,
    Excess is Liquid - C1,
    adjust(Liquid, Excess, W1,W2).
update(jugs(V1,V2), transfer(1,2), jugs(W1,W2)) :-
    capacity(2,C2),
    Liquid is V1 + V2,
    Excess is Liquid - C2,
    adjust(Liquid, Excess, W2,W1).

adjust(Liquid, Excess, Liquid, 0) :- Excess <= 0.
adjust(Liquid, Excess, V, Excess) :- Excess > 0, V is Liquid - Excess.

```

```
legal(jugs(_,_)).
```

```
capacity(1,8).
```

```
capacity(2,5).
```

### 1.2.2 Тесты

```
?- test_dfs(X).
```

```
X = [fill_and_transfer(1), fill_and_transfer(1), fill_and_transfer(1)]
```

```
?-
```