

# Решение задачи про сортировку жидкостей

@medvezhonokok

1 декабря 2025 г.

Для удобства, введем несколько определений:

- $\mathcal{F}$  - множество всех колб
- $a \Rightarrow b$  - возможность перелить жидкость из колбы  $a$ , в колбу  $b = \{0, 1\}$
- $|f|$  - количество капель в колбе  $f$

Тогда задача не имеет решения, если: не существует колбы, из которой можно перелить жидкость в любую другую колбу:

$$\nexists f \in \mathcal{F} : \nexists g \in \mathcal{F} : g \neq f : (f \Rightarrow g) \equiv 1$$

ТЗ не требует оптимального решения  $\Rightarrow$  идем жадным алгоритмом. Запоминаем каждый шаг (*Move*  $m \{from, to, color\}$ ) в *Set*<*Move*> *moves*. Несложно заметить, что если в *moves* уже есть текущий шаг  $\Rightarrow$  цикл  $\Rightarrow$  этот шаг не приведет к ответу. Давайте перебирать все **хорошие** шаги, пока решение есть (*can\_solve*) или хорошие шаги не кончатся.

```
1: procedure GETGOODMOVES( $\mathcal{F}$ , parent)
2:   moves  $\leftarrow [ ]$ 
3:   for  $i \leftarrow 0$  до  $|\mathcal{F}| - 1$  do
4:      $f \leftarrow \mathcal{F}_i$ 
5:     if ISEMPTY( $f$ ) then continue
6:     end if
7:     for  $j \leftarrow 0$  до  $|\mathcal{F}| - 1$  do
8:       if  $i = j$  then continue
9:       end if
10:       $g \leftarrow \mathcal{F}_j$ 
11:      if ISFULL( $g$ ) then continue
12:      end if
13:      if  $f \Rightarrow g \equiv 1$  then
14:        moves  $\leftarrow moves \cup \{(i, j, \text{LAST}(f))\}$ 
15:      end if
16:    end for
17:  end for
```

```

18:   return moves
19: end procedure

```

Давайте перебирать все **хорошие** шаги, пока решение есть (*can\_solve*) или **хорошие** шаги не кончатся.

```

1: procedure CANSOLVE( $\mathcal{F}$ )
2:   for  $i \leftarrow 0$  до  $|\mathcal{F}| - 1$  do
3:      $f \leftarrow \mathcal{F}_i$ 
4:     if ISEMPTY( $f$ ) then continue
5:     end if
6:     for  $j \leftarrow 0$  до  $|\mathcal{F}| - 1$  do
7:       if  $i = j$  then continue
8:       end if
9:        $g \leftarrow \mathcal{F}_j$ 
10:      if  $f \Rightarrow g \equiv 1$  then
11:        return true
12:      end if
13:    end for
14:  end for
15:  return false
16: end procedure

```

Дальше построим граф шагов, для начального состояния получаем список **хороших шагов**, высчитываем состояние  $\mathcal{F}_i$  после применения  $i$ -го шага, если для  $\mathcal{F}_i$  есть следующие хорошие шаги, то повторяем алгоритм пока задача не решится, если мы понимаем, что задача не может быть решена, то поднимаемся до родителя  $i$ -го шага.

```

1: procedure GETSOLUTION(flasks, root, visited)
2:   if ISOLVED(flasks) then
3:     return true
4:   end if
5:    $state \leftarrow \text{GETSTATEHASH}(\textit{flasks})$ 
6:   if  $state \in \textit{visited}$  then // cycle!
7:     return false
8:   end if
9:    $\textit{visited} \leftarrow \textit{visited} \cup \{state\}$ 
10:  for  $move \in \text{GETGOODMOVES}(\textit{flasks}, \textit{root})$  do
11:    COMMIT( $move$ , flasks)
12:    ADDCHILD(root,  $move$ )
13:    if CANSOLVE(flasks,  $move$ , visited) then
14:      return true
15:    end if

```

```

16:         ROLLBACK(move, flasks)
17:         REMOVECHILD(root, move)
18:     end for
19:     visited  $\leftarrow$  visited  $\setminus$  {state}
20:     return false
21: end procedure

```