

16/03/2018 Вершинный поиск: программа поиска.

В примерах ниже подразумевается граф  $G = (VG, EG)$

```
test3.graph
0: 2 3 6 14
1: 6 7 11
2: 0 7 11 14
3: 0 6 12
4: 6 11 13
5:
6: 0 1 3 4 9 12
7: 1 2
8: 13
9: 6 13 14
10:
11: 1 2 4 13
12: 3 6
13: 4 8 9 11
14: 0 2 9
```

Пусть  $\Pi = \{Z_0, \dots, Z_m\}$  — программа преследователей в задаче вершинного поиска на  $G$ . Допустимыми считаются программы, удовлетворяющие:

1.  $Z_i \subseteq VG$ ,  $Z_0 = \emptyset$ ;
2. на каждом шаге разрешается либо ставить новых игроков, либо снимать уже поставленных;
3. допустимо ставить только одного дополнительного игрока;
4. снимать разрешается любое число игроков.

3.1 Каждый шаг программы (в виде списка занятых на данном шаге вершин, заключенных в квадратные скобки) записан с новой строки. Проверить, является ли  $\Pi$  допустимой. Если да, то вычислить, какое наибольшее число игроков одновременно размещено в вершинах графа. Если программа содержит недопустимый шаг, указать, какое из требований не выполняется (через запятую, если нарушено несколько пунктов): '-1', если  $Z_i \not\subseteq VG$  или  $Z_0 \neq \emptyset$ ; '-2', если произошло одновременное снятие игроков и их постановка; '-3', если поставлено более одного игрока.

```
test3.in
#3.1
[ ]
[3]
[0, 3]
[0, 3, 12]
[0, 12]
[0, 6, 12]
[0, 4, 6, 12]
[0, 6, 12]

test3.out
#3.1
4
```

3.2 Проверить, является ли программа выигрывающей. Если да, то указать 'Y', если нет, то отобразить очищенный подграф на момент последнего хода игроков для рассматриваемой П (в принятом нами формате записи графа).

*test3.out*

#3.2

0: 6

12: 6

6: 0 12

3.3 Проверить, является ли программа монотонной (да — '1', нет — '0').

*test3.out*

#3.3

0