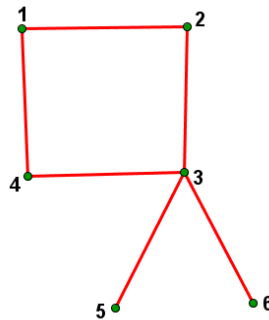


## ЛЕКЦИЯ 11. ОБХОДЫ ГРАФА

### МАШИННОЕ ПРЕДСТАВЛЕНИЕ ГРАФОВ



### МАТРИЦА СМЕЖНОСТИ

---

Строки и столбцы – вершины ( $p \times p$ ).

Для неориентированного графа – симметричная, для ориентированного – нет.

Если есть петли – на диагонали 1. Для мультиграфа числа могут быть  $>1$ .

### МАТРИЦА ИНЦИДЕНТНОСТИ (ИНЦИДЕНЦИЙ)

---

Строки соответствуют вершинам, столбцы – рёбрам ( $p \times q$ ).

Если вершина и ребро инцидентны, то 1, иначе – 0.

Для обычного неориентированного графа в каждом столбце ровно две 1. Если граф ориентирован, то ставится -1 и 1. Если есть петли, то будет одна 1. Если мультиграф, то могут быть числа  $>1$ .

### СПИСКИ СМЕЖНОСТИ

---

Массив списков.

Длина массива = число вершин.

$i$ -й элемент – список вершин, смежных с вершиной  $i$ .

### ГЕНЕРАТОРЫ СПИСКОВ СМЕЖНОСТИ

---

Процедура генерирует список смежности «на лету».

Пр и м е р : игра в «пятнашки».

### ОБХОД В ГЛУБИНУ (DFS - DEPTH-FIRST SEARCH)

ОБХОД:  $1 \downarrow 2 \downarrow 3 \downarrow 4$  ,  $\uparrow 3$  ,  $\downarrow 5$  ,  $\uparrow 3$  ,  $\downarrow 6$  ,  $\uparrow 3 \uparrow 2 \uparrow 1$

A – матрица смежности

n – число вершин

# В начале все вершины не помечены

Mark = [False]\*n

# Рекурсивная процедура обхода в глубину

```
def DFS(u):
    global A, Mark
    Mark[u] = True
    print(u+1, end = " ")
    for v in range(n):
        if A[u][v]==1 and not Mark[v]:
            DFS(v)
DFS(source-1)
print()
```

## ПРИМЕНЕНИЯ ОБХОДА В ГЛУБИНУ

---

1. Связность и компоненты связности
2. Путь (не кратчайший!)
3. Поиск мостов и точек сочленения

## ОБХОД В ШИРИНУ (BFS - BREADTH-FIRST SEARCH)

ОБХОД: 1, 1-й ярус:2,4, 2-й ярус: 3, 3-й ярус:5,6

```
A – матрица смежности
n – число вершин
# В начале все вершины не помечены
Mark = [False]*n
# Инициализация очереди
Q = [source-1]
Mark[source-1] = True
head = 0
# Обход в ширину
while head<len(Q):
    u = Q[head]
    print(u+1, end = " ")
    for v in range(n):
        if A[u][v]==1 and not Mark[v]:
            Mark[v] = True
            Q.append(v)
    head += 1
print()
```

## **ПРИМЕНЕНИЯ ОБХОДА В ШИРИНУ**

---

1. Связность и компоненты связности
2. Расстояния
3. Ярусы
4. Путь (кратчайший)

## **СЛОЖНОСТЬ ОБХОДОВ В ГЛУБИНУ И ШИРИНУ**

При правильной реализации сложность =  $O(p + q)$