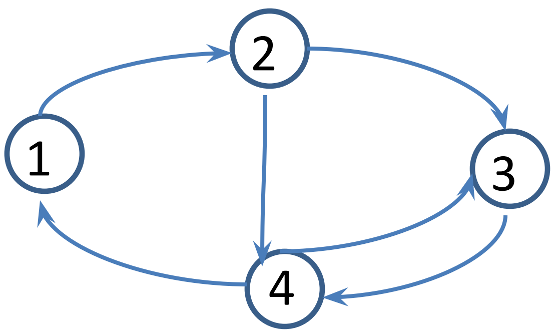
Представление графов

Лекция 3

Матрица смежностей

Пусть дан граф *G*= (*V*,*E*), *N* = |*V*|, *M* = |*E*|.

Матрица смежностей для графа G – это матрица *A* размера *N*x*N*, состоящая из 0 и 1, в которой *A*[*i*, *j*]=1 тогда и только тогда, когда есть ребро из узла *i* в узел *j*.



	1	2	3	4
1	0	1	0	0
2	0	0	1	1
3	0	0	0	1
4	1	0	1	0

Матрица инцидентностей

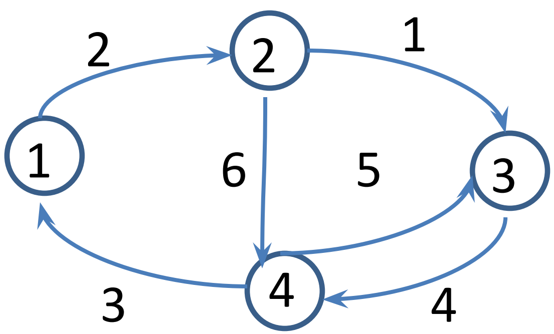
Матрица инцидентностей *для графа G – это* матрица *B* размера *N*x*M*, в которой :

B[*i*, *j*]=

1, если ребро j инцидентно вершине *i*,

-1, если ребро j входит в вершину *i*,

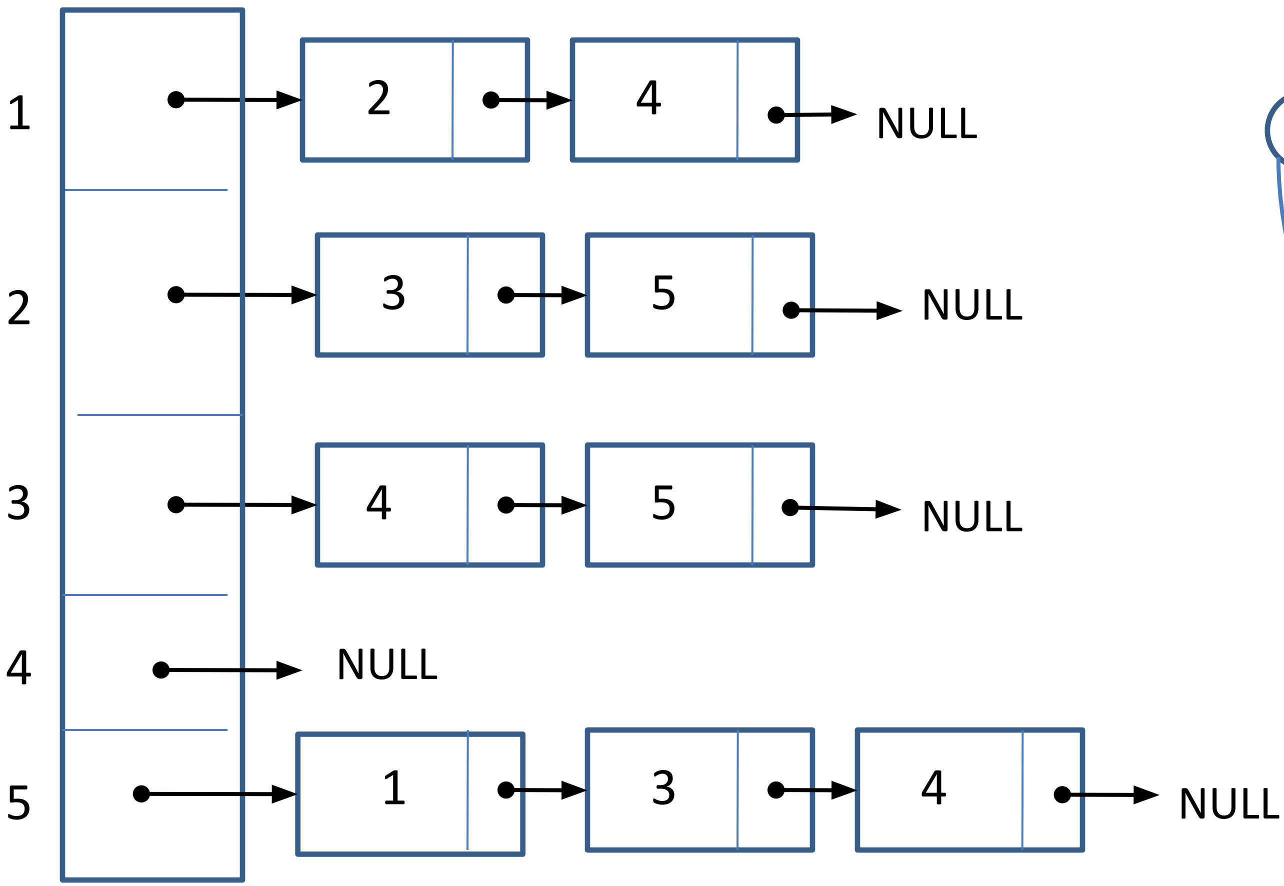
0, если ребро j не связано с вершиной *i*.

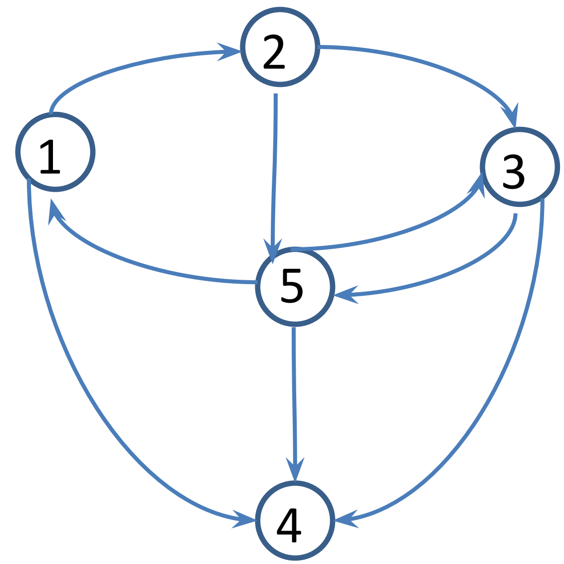


	1	2	3	4	5	6
1	0	1	-1	0	0	0
2	1	-1	0	0	0	1
3	-1	0	0	1	-1	0
4	0	0	1	-1	1	-1

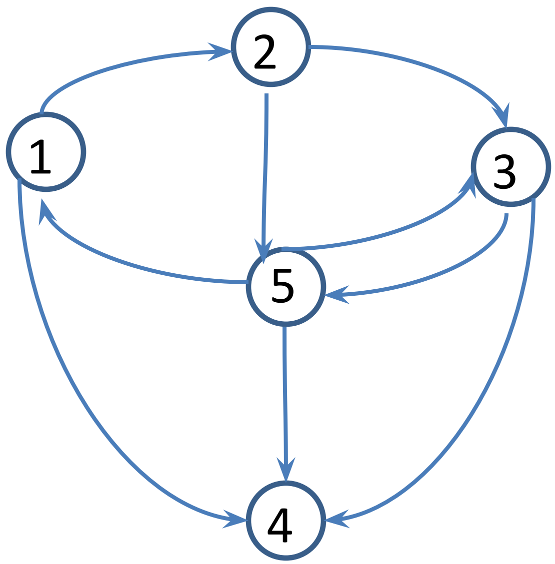
Списки смежностей

Списком смежностей для узла *v* называется список всех узлов *w*, смежных с *v*.





Табличное представление списков смежностей



	Номер вершины	Следующий
1	1	6
2	2	8
3	3	10
4	4	0
5	5	12
6	2	7
7	4	0
8	3	9
9	5	0
10	4	11
11	5	0
12	1	13
13	3	14
14	4	0

Топологическая сортировка

Определение. *Частичным порядком* на множестве A называется отношение R, определенное на A и такое, что

- R транзитивно,
- для всех $a \in A$ утверждение aRa ложно, т.е. отношение R иррефлексивно.

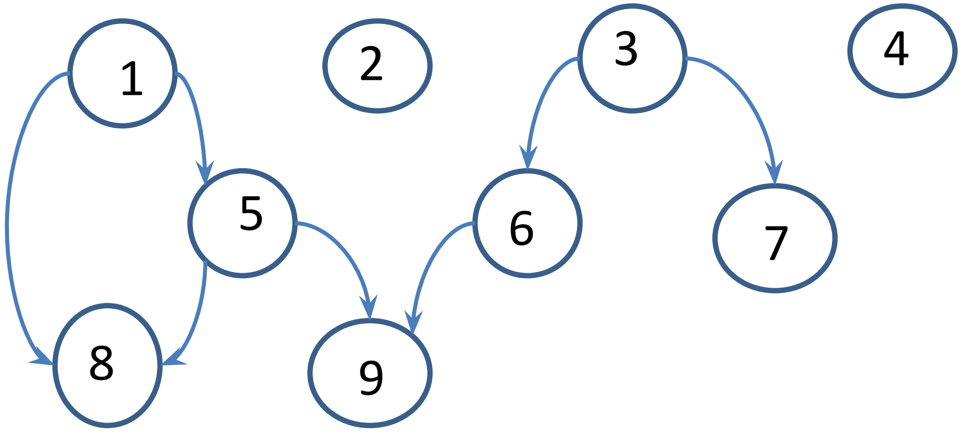
Из свойств (1) и (2) следует, что если aRb истинно, то bRa ложно (асимметричность).

Примеры частичного порядка:

- решение большой задачи разбивается на ряд подзадач, над которыми установлен частичный порядок: без решения одной задачи нельзя решить несколько других;
- последовательность чтения курсов в учебных программах: один курс основывается на другом;
- выполнение работ: одну работу следует выполнить раньше другой.

Если R — частичный порядок на множестве A, то (A, R) — ациклический граф.

Если (A, R') — ациклический граф и R' — отношение "являться потомком", определенное на A, то R' — частичный порядок на A.



Определение. *Линейный порядок R* на множестве A — это такой частичный порядок, что если a и b принадлежат A, то либо aRb, либо bRa, либо $a = b$.

Если A — конечное множество, то линейный порядок R удобно представлять , считая все элементы множества A расположенными в виде последовательности

$$a_1, a_2, \dots, a_n,$$

для которой имеет место $a_i R a_j$ тогда и только тогда, когда $i < j$.

Если задан частичный порядок *R* на множестве *A*, часто бывает нужен линейный порядок, содержащий этот частичный порядок.

Эта проблема вложения частичного порядка в линейный называется *топологической сортировкой.*

Формально можно сказать, что

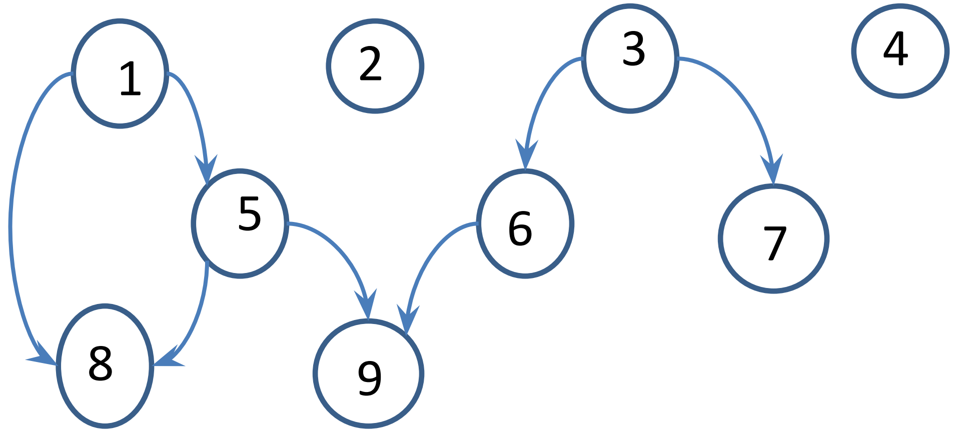
частичный порядок *R* на множестве *A*

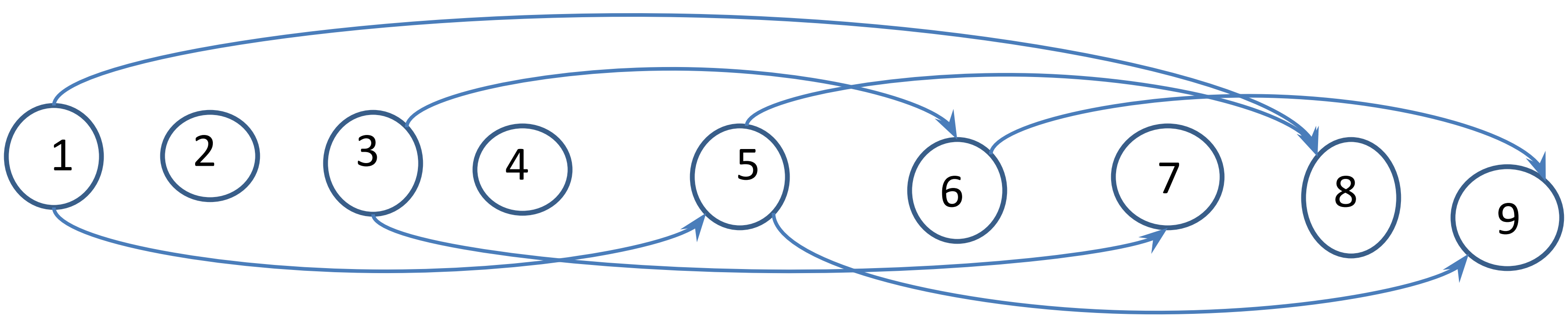
вложен в линейный порядок *R'*,

если *R*‘ — линейный порядок и $R \subseteq R'$,

т. е. *aRb* влечет *aR'b* для всех а и *b* из *A*.

Топологическая сортировка. Пример





Алгоритм. Топологическая сортировка

Вход. Частичный порядок R на конечном множестве *A.*
Выход. Линейный порядок R' на *A, для* которого $R \subseteq R'$.

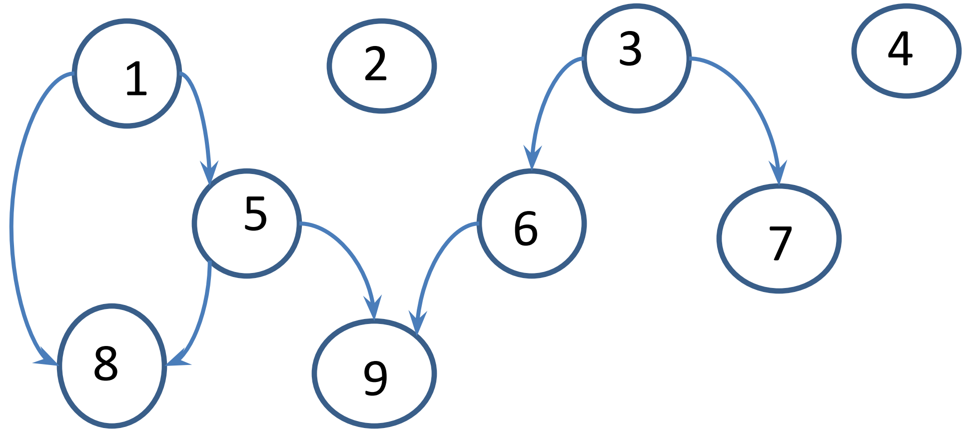
Метод. Так как *A* — конечное множество, линейный порядок R' на *A* можно представить в виде списка $a_1, a_2, \dots, a_n$, для которого
$a_i \, R' \, a_j$, если $i < j$, и $A = \{a_1, a_2, \dots, a_n\}$.
Эта последовательность элементов строится с помощью следующих шагов:

(1) Положить $i=1$, $A_i=A$ и $R_i=R$.

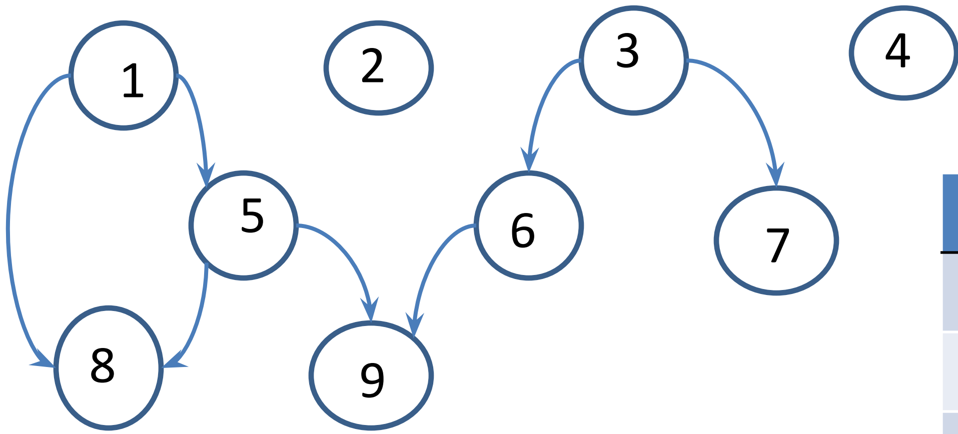
(2) Если A_i пусто, остановиться и выдать $a_1, \dots, a_i$, в качестве искомого линейного порядка. В противном случае выбрать в A_i такой элемент a_{i+1} что $a' \, R \, a_{i+1}$, ложно для всех $a' \in A_i$.

(3) Положить $A_{i+1} = A_i \setminus \{a_{i+1}\}$ и $R_{i+1} = R_i \setminus (\{a_{i+1}\} \times A_{i+1})$. Затем увеличить *i* на единицу и повторить шаг 2.

Топологическая сортировка. Пример



Топологическая сортировка. Реализация на матрице смежности

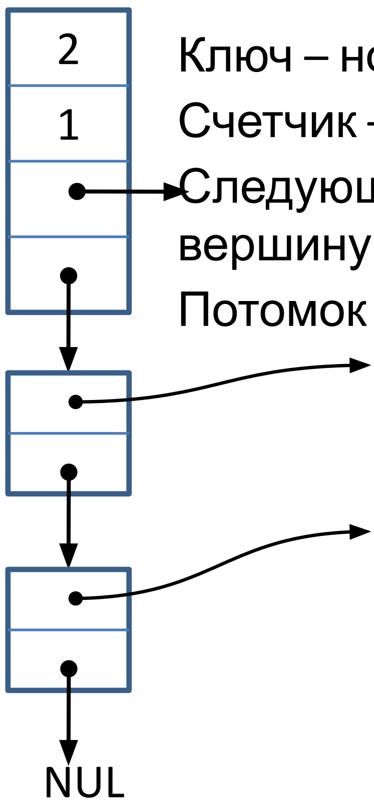


1. Найти вершину, в которую не входит ни одна дуга (это нулевой столбец).
   Удалить все выходящие из нее дуги (обнулить соответствующую строку)
2. Пока не перебрали все вершины, повторять шаг 1.

	1	2	3	4	5	6	7	8	9
1	0	0	0	0	1	0	0	1	0
2	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	1	0	0
4	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	1	1
6	0	0	0	0	0	0	0	0	1
7	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0

Топологическая сортировка. Реализация на иерархических списках

Элемент списка вершин графа:



Ключ – номер вершины

Счетчик – количество входящих дуг

Следующий – указатель на следующую вершину

Потомок – список указателей на потомков

Работа алгоритма(построение)

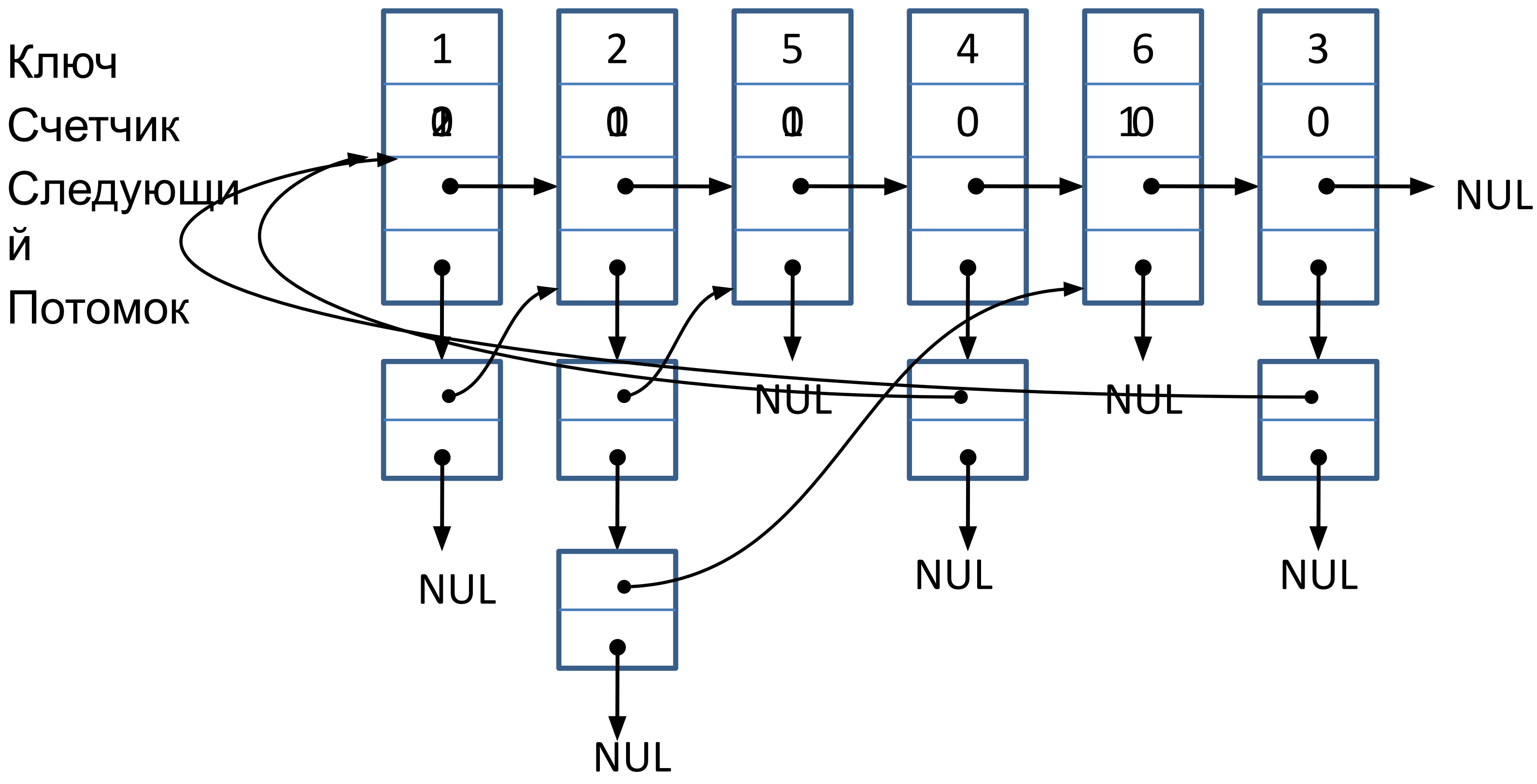
1< 2;

2< 5;

4 < 1;

2 < 6;

3 < 1;



Работа алгоритма (перестройка списка)