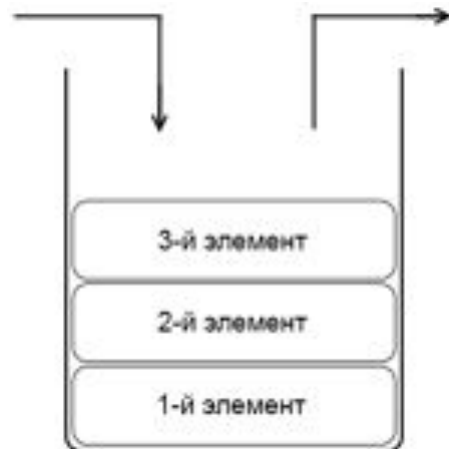


Цель лекции: дать основные понятия структур данных.

Структура данных – это способ хранения и организации данных, облегчающий доступ к этим данным и их модификацию.

Элементарные структуры данных

Стек (англ. **stack** — **стопка**) — структура данных с методом доступа к элементам LIFO (англ. Last In — First Out, «последним пришел — первым вышел»).



Элементарные структуры данных

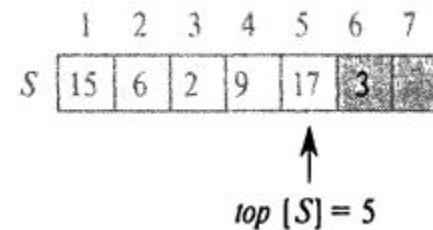
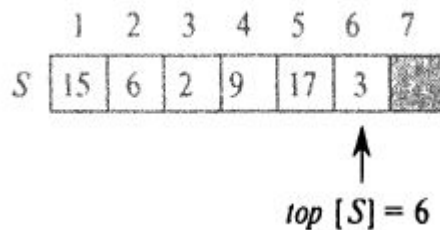
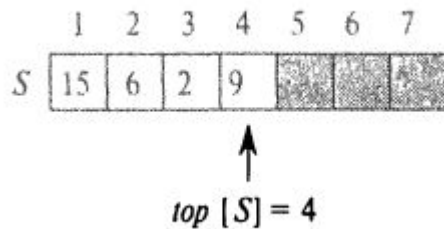
Основные операции:

- инициализация (*Init*)
- деструктуризация (*Destroy*)
- помещение элемента в стек (*Push*)
- удаление элемента из стека (*Pop*)
- значение верхнего элемента (*Top*)
- проверка на пустоту (*isEmpty*)
- проверка на полноту (*isFull*)

```
PUSH(S, x)
1  top[S] ← top[S] + 1
2  S[top[S]] ← x
```

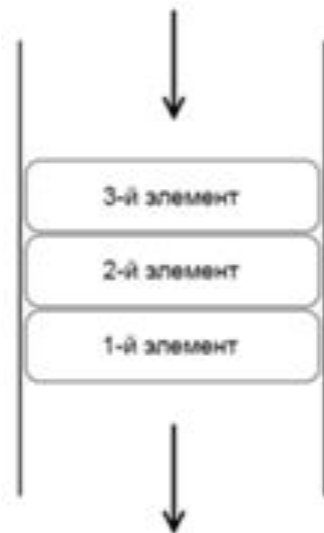
```
POP(S)
1  if STACK_EMPTY(S)
2     then error "Underflow"
3     else top[S] ← top[S] - 1
4         return S[top[S] + 1]
```

Элементарные структуры данных

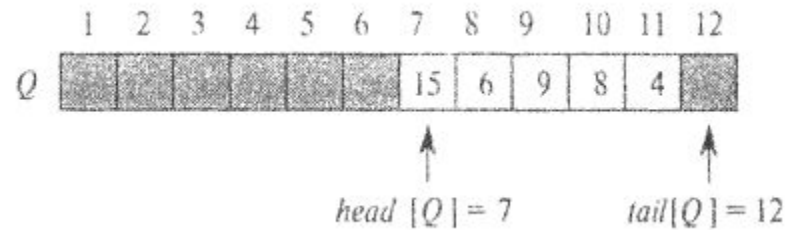


Элементарные структуры данных

Очередь (англ. queue) — структура данных с методом доступа к элементам по принципу FIFO (First In First Out), «первым пришел — первым вышел»).



Элементарные структуры данных



Элементарные структуры данных

Основные операции:

- инициализация (*Init*)
- деструктуризация (*Destroy*)
- помещение элемента в очередь (*ENQUEUE*)
- удаление элемента из очереди (*DEQUEUE*)
- значение первого элемента (*Head*)
- значение последнего элемента (*Tail*)
- проверка на пустоту (*isEmpty*)
- проверка на полноту (*isFull*)

ENQUEUE (*Q*, *x*)

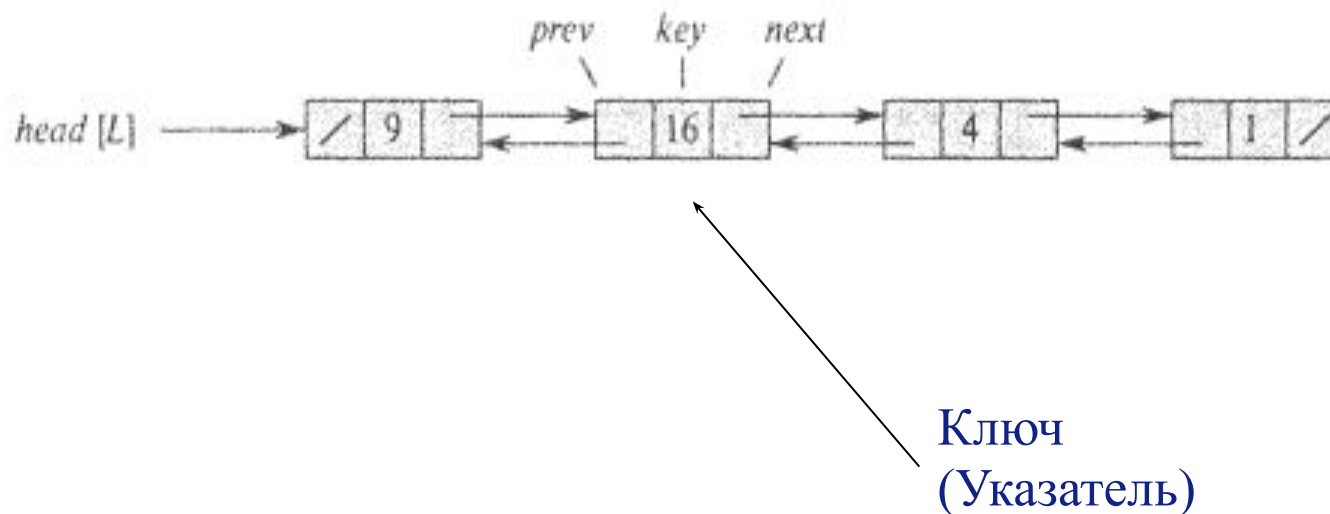
```
1  Q[tail[Q]] ← x
2  if tail[Q] = length[Q]
3      then tail[Q] ← 1
4      else tail[Q] ← tail[Q] + 1
```

DEQUEUE (*Q*)

```
1  x ← Q[head[Q]]
2  if head[Q] = length[Q]
3      then head[Q] ← 1
4      else head[Q] ← head[Q] + 1
5  return x
```

Элементарные структуры данных

Связанный список (linked list) — это структура данных, в которой объекты расположены в линейном порядке.



Элементарные структуры данных

Основные операции

Поиск в связанном списке LIST_SEARCH(L, k)

```
LIST_SEARCH(L, k)
1  x ← head[L]
2  while x ≠ NIL and key[x] ≠ k
3      do x ← next[x]
4  return x
```

Вставка в связанный список LIST_INSERT(L, x)

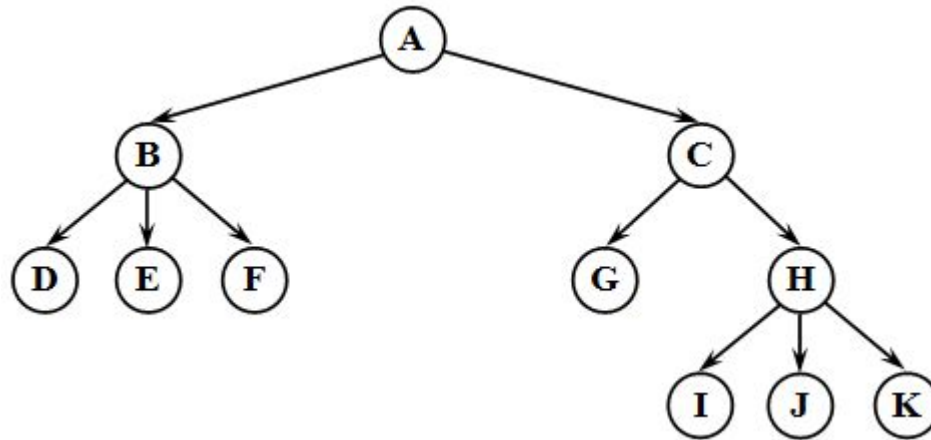
```
LIST_INSERT(L, x)
1  next[x] ← head[L]
2  if head[L] ≠ NIL
3      then prev[head[L]] ← x
4  head[L] ← x
5  prev[x] ← NIL
```

Удаление из связанного списка LIST_DELETE(L, x)

```
LIST_DELETE(L, x)
1  if prev[x] ≠ NIL
2      then next[prev[x]] ← next[x]
3      else head[L] ← next[x]
4  if next[x] ≠ NIL
5      then prev[next[x]] ← prev[x]
```

Нелинейные структуры данных

Дерево – это структура данных, представляющая собой совокупность элементов и отношений, образующих иерархическую структуру этих элементов .



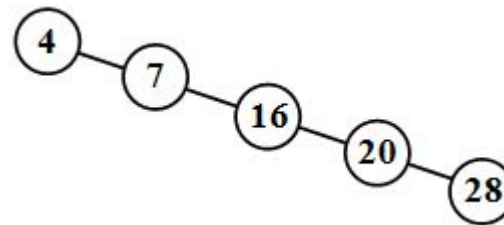
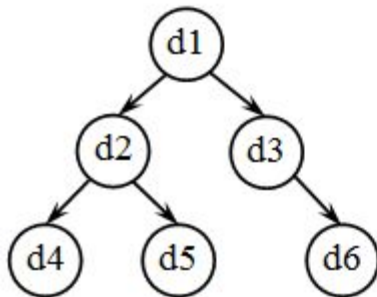
Каждое дерево обладает следующими свойствами:

- существует узел, в который не входит ни одной дуги (корень);
- в каждую вершину, кроме корня, входит одна дуга.

Нелинейные структуры данных

Бинарное (двоичное) дерево – это динамическая структура данных, представляющее собой дерево, в котором каждая вершина имеет не более двух потомков .

Частный случай бинарного дерева – список



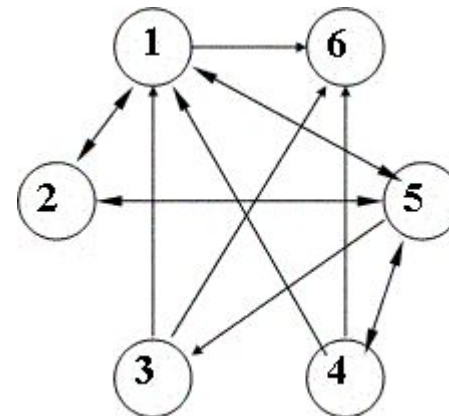
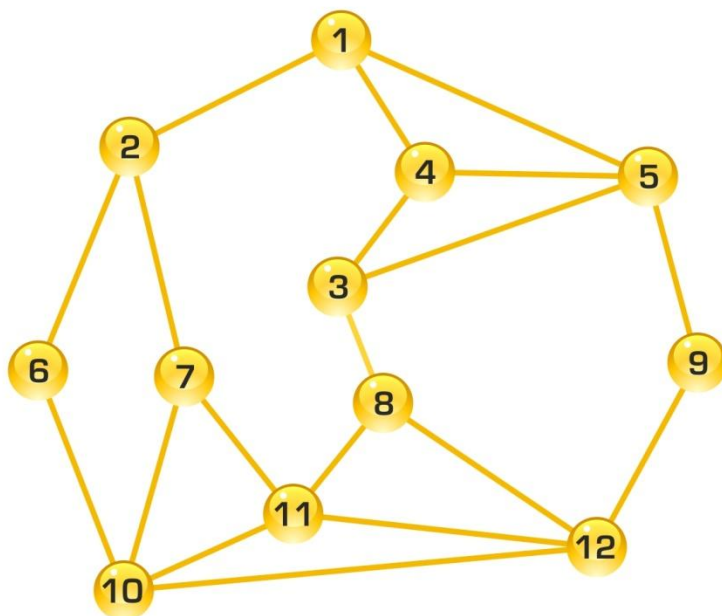
Основные операции

- создание бинарного дерева;*
- печать бинарного дерева;*
- обход бинарного дерева;*
- вставка элемента в бинарное дерево;*
- удаление элемента из бинарного дерева;*
- проверка пустоты бинарного дерева;*
- удаление бинарного дерева.*

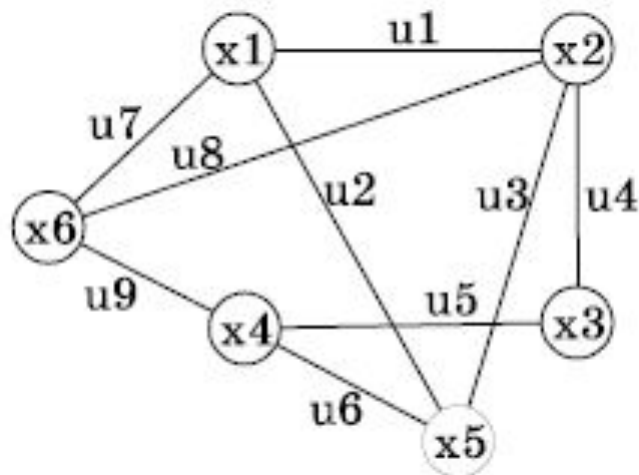
Нелинейные структуры данных

Понятие *графа* опирается на понятие *множества*. Графом можно назвать объект, состоящий из двух множеств: точек и линий, которые находятся между собой в некотором отношении. Множество точек графа обозначается $X = \{x_1, x_2, \dots, x(n)\}$, $|X|=n$, и называется множеством вершин. Множество линий, соединяющих любые пары вершин, $(x(i), x(j)) \in X$ называется множеством ребер или дуг и обозначается $U = \{u_1, u_2, \dots, u(m)\}$, $|U|=m$.

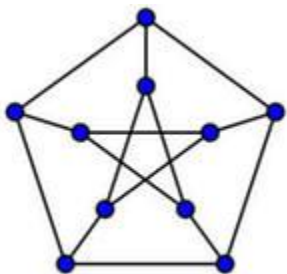
Тогда графом можно считать математический объект, который обозначается $G=(X,U)$ и состоит из множества вершин и множества ребер или дуг, находящихся между собой в некотором отношении.



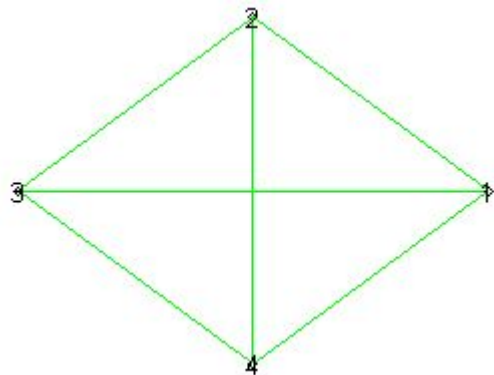




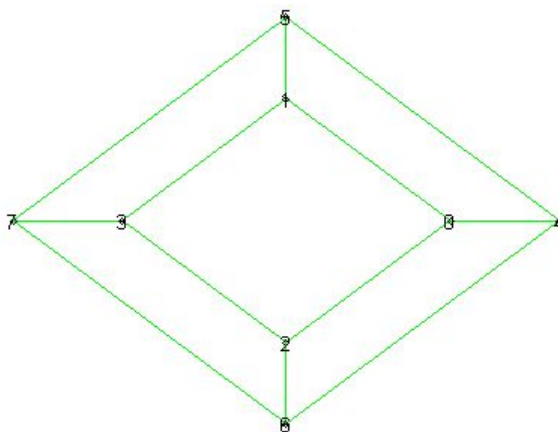
Специальные графы
Граф Петерсона



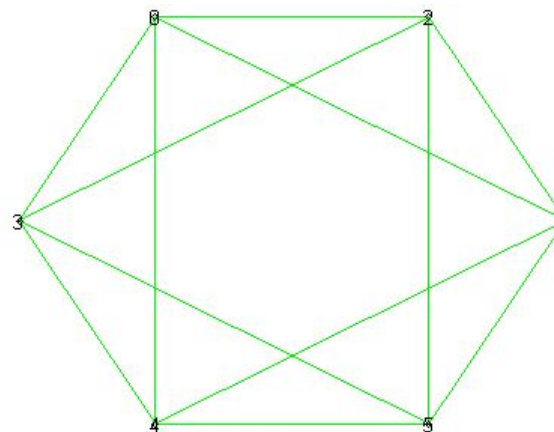
Платоновы графы
Тетраэдр



Куб



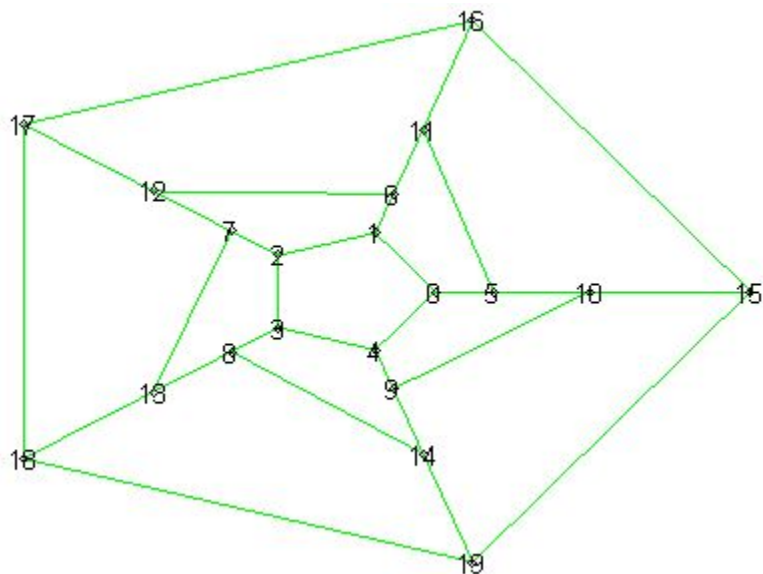
Октаэдр



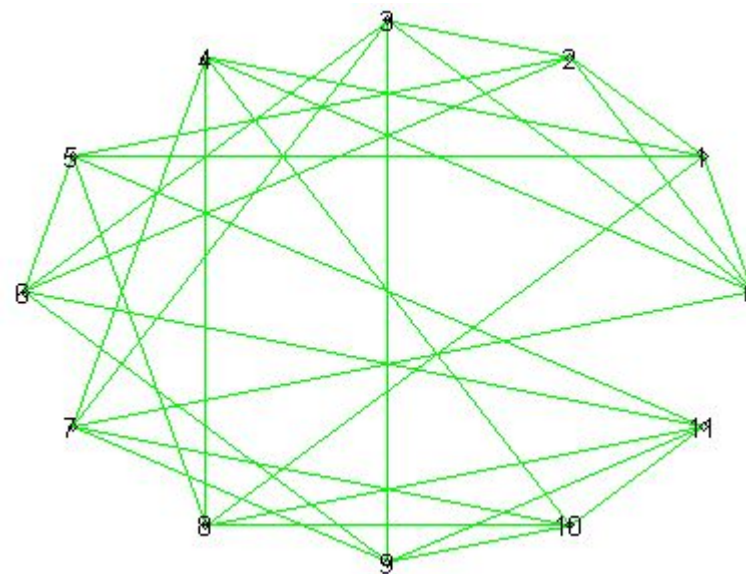
Специальные графы

Платоновы графы

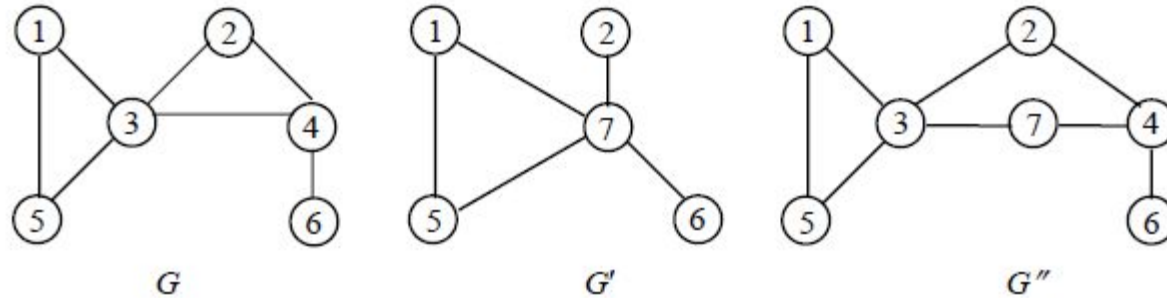
Додекаэдр



Икосаэдр

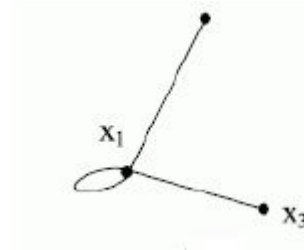
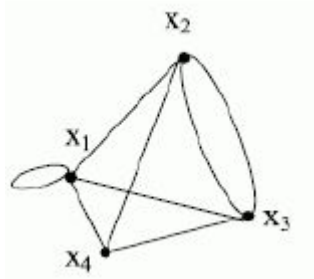


Операции над графами
-Локальные

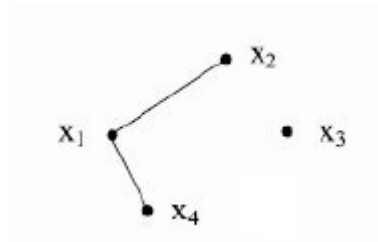


-Граф

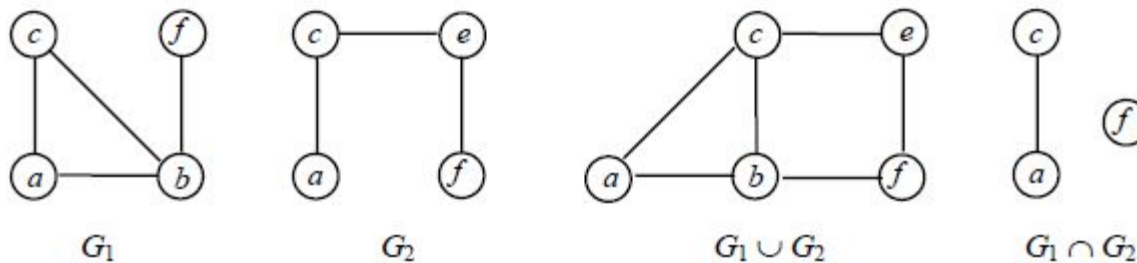
Подграф



-Суграф



- алгебраические



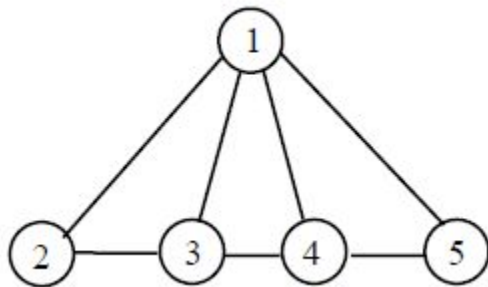
Маршруты, цепи, циклы

Маршрут в графе – это последовательность вершин x_1, x_2, \dots, x_n , такая, что для каждого $i = 1, 2, \dots, n - 1$ вершины x_i и x_{i+1} соединены ребром.

Маршрут, в котором нет повторяющихся ребер, называется цепью.

Замкнутая цепь называется циклом.

Цепь и цикл называются простыми, если нет повторяющихся вершин.



Последовательность вершин:

2, 3, 5, 4 – не маршрут

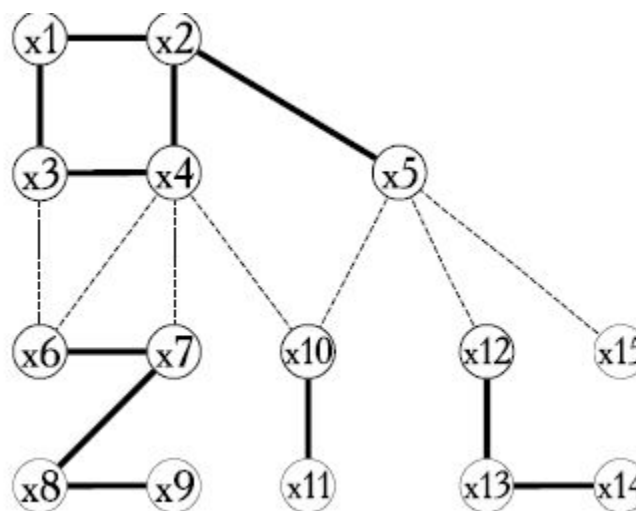
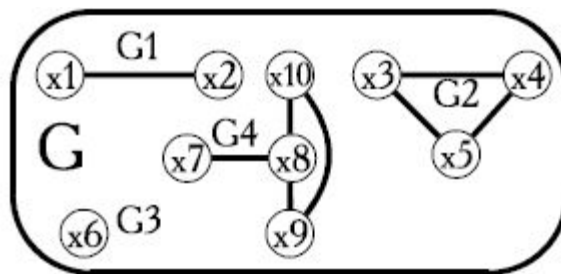
2, 3, 4, 5, 1, 4, 3 – маршрут, но не цепь

3, 1, 4, 5, 1, 2 – цепь, но не простая

2, 3, 1, 4, 3, 1, 2 – маршрут, но не цикл

2, 3, 1, 4, 5, 1, 2 – цикл, но не простой

2, 3, 4, 5, 1, 2 – простой цикл

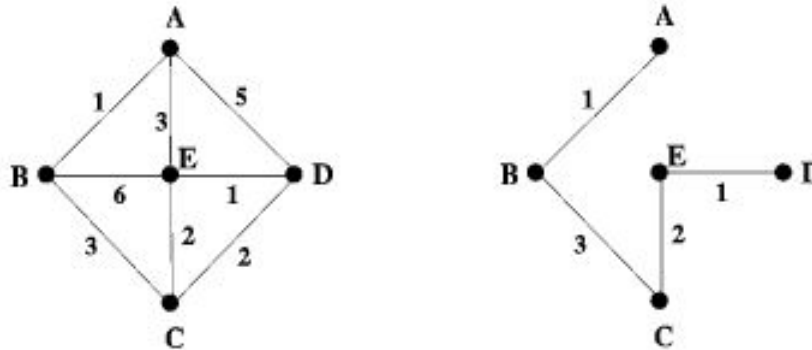


Рассмотрим метрику и основные числа графов. Метрика графа основана на понятии расстояния. Расстоянием $d(x(i), x(j))$, между вершинами $x(i), x(j) \in X$ графа $G=(X, U)$ называется длина кратчайшей цепи, соединяющей эти вершины. Под длиной цепи понимается число входящих в нее ребер. Функция $d(x(i), x(j))$, определенная на множестве ребер графа G , называется метрикой графа.

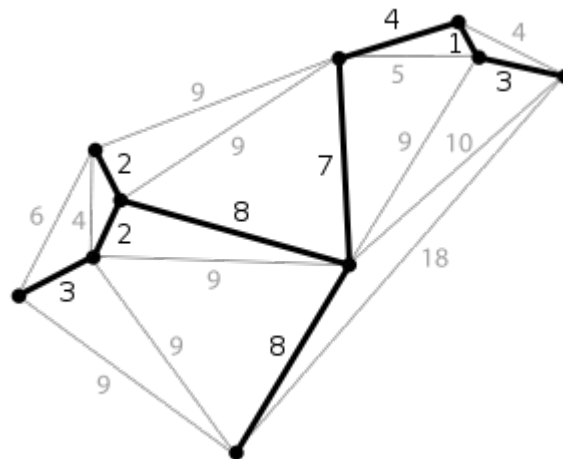
Связанный граф без циклов называется деревом.

Теорема

Дерево, у которого число вершин равно числу вершин графа, из которого выделено это дерево, называется покрывающим деревом.



Минимальное остовное дерево (или минимальное покрывающее дерево) в связанном, взвешенном, неориентированном графе — это покрывающее дерево этого графа, имеющее минимальный возможный вес, где под весом дерева понимается сумма весов входящих в него рёбер.



Цикломатическое число графа

Наименьшее число ребер, которое необходимо удалить из графа G , чтобы он стал ациклическим (деревом), называется цикломатическим числом графа.

Основные выводы:

1. Изучены основные понятия структур данных;
2. Рассмотрено применение теории при решении задач конструкторско-технологической информатики;