Алгоритмы и структуры данных

**Массив** - это структура однотипных данных, занимающих непрерывную область памяти. Массив имеет размер кол-во эл в нем, каждый эл. имеет свой номер - индекс, обращение к элементу массива осуществляется к указанию его индекса

Одномерный массив (вектор) - содержит только одну строку данных, поэтому к эл хранящемуся в массиве довольно просто получить доступ с помощь одного числового индекса.

**Двумерный массив (многомерный**) - содержит более 1 строки с данными, поэтому его индекс будет состоять из пары чисел, одно из которых идентифицирует строку, а другой столбец, короче матрица.

Если в процессе выполнения программы требуется изменять размер массива, то в этом случае первоначальный массив объявляют как динамический  
Массивы границы, которых явно заданы в описании - **статические**, их размер известен и не меняется по ходу программы

**Алгоритмы сортировок:** (см. Презентацию 2)

Пузырьком - на каждом шаге мы находим наибольший эл из двух соседних и ставим этот эл в конец пары, получается что при каждом прогоне цикла большие эл будут всплывать к концу массива.

Сортировка вставкой**:**

1 шаг - 2 эл массива сравнивается с эл который стоит перед ним, те 1, если 2 эл меньше 1, то 2 эл вставляется в позицию 1 эл

2 шаг- 3 эл массива сравнивается с эл который появляется перед ним, если этот эл меньше 1, то он вставляется на позицию 1 и 1,2 сдвигаются вправо. Если 3>1 но 3<2, то он вставляется на позицию второго. Если 3 больше обоих то он остается на том же месте

3 шаг- если есть n эл в сортировке то эта процедура повторяется n-1 раз чтобы получился отсортированный список массивов

Сортировка выбором - алгоритм сортировки выбором заключается в поиске на необработанном срезе массива мин значения и в дальнейшем обмене этого значения с первым эл необработанного среза. На след шаге необработанный срез уменьш на 1 эл, продолжать выполнять поиск и обмен пока не будет достигнут конец списка или массив.

Сортировка Шелла -идея метода заключается в сравнении разделении

Сортировка Хоара (быстрая) -

Алгоритмы поиска: (см. Презентацию 2)

Бинарный поиск значения в массиве используется только для отсортированных по возр или убыв последовательностей. Заключается в определении содержит ли массив искомое значение, а также в определении места его нахождения

1 шаг - находится средний эл последовательности для этого первый и последний связываются с переменными, а индекс среднего эл вычисляется

2 шаг - значение среднего эл сравнивается с искомым значением в зависимости от того больше оно или меньше значения среднего эл дальнейший поиск будет происходить только в левой или правой половины массива. Если значение среднего эл оказывается равным искомому то поиск завершается

3 шаг - иначе одна из границ исследуемой последовательности сдвигается. Если искомое значение больше среднего эл то нижняя граница сдвигается за средний эл на один эл справа. Если иск значение меньше среднего эл то внешняя граница сдвигается на эл перед средней.

4 шаг - снова находится средний эл теперь уже в выбранной половине и описанный выше алгоритм повторяется до иск значения

Далее слайды про работу в Python с массивами