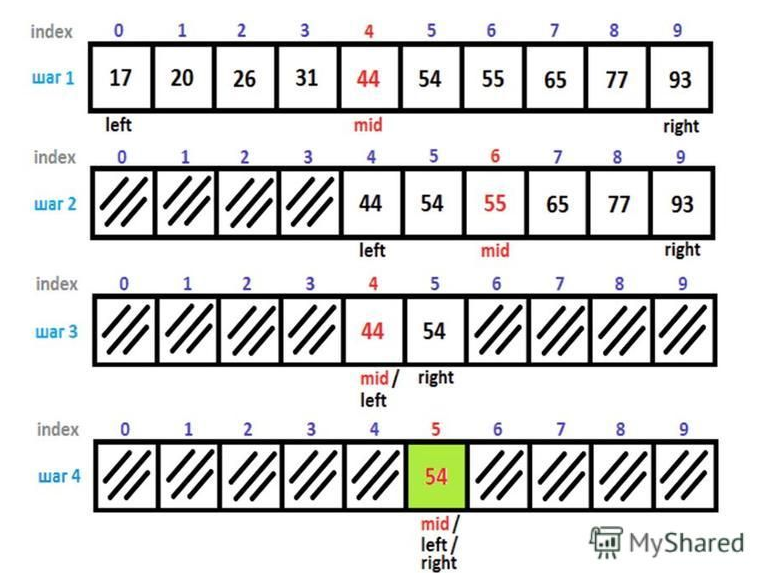
**Методы поиска:**

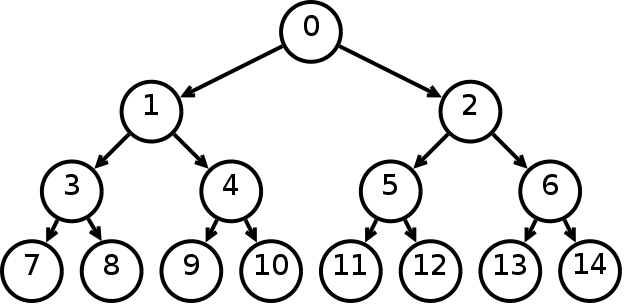
* **Бинарный поиск; Сложность – O (log n)**

В отсортированном массиве находится середина и сравнивается с искомым элементом, если искомое число больше, то берется правая часть массива, если меньше, то левая часть. И так до тех пор пока не найдется искомый элемент.

****

* **Бинарное дерево**

Используется в основном для добавления данных в уже отсортированный массив. Это структура данных, где элементы отсортированы от верхнего элемента к нижнему с условием, что слева от родительского элемента вниз по списку идут элементы меньше родительского, а справа – больше.



* **Фибоначчиев поиск; Сложность – O (ln n)**
* **Интерполяционный поиск; Сложность – O (log log n)**

Похож на поиск в словаре или телефонной книге. Предсказывает нахождение элемента, по расстоянию между ключом и элементом. Средний элемент выбирает по формуле:

mid = left + (key - a[left]) \* (right - left) / (a[right] - a[left])



**Разрешение коллизий хеширования:**

* **Простое рехеширование (Hash Tables)**

Хэш-таблица – это таблица, в которой индексом элемента служит его хэш значение. Поиск по этой таблице можно осуществлять по значению элемента, либо по его хэшу.

Обычно в хэш таблицах хэш функция привязана к количеству элементов в таблице, поэтому, когда в таблицу нужно добавить элемент или удалить его приходится создавать новую таблицу – это называется Рехеширование (заново хешировать):

* + Метод открытой адресации – при коллизии, второй элемент ставится через одну ячейку после занятой ячейки. Найти это значение можно перебирая ячейки после первого совпадения, до пустой ячейки.
* **Рехеширование с помощью псевдослучайных чисел**
* **Метод цепочек**

Метод цепочек – вместо одной ячейки напротив хэш индекса будет стоять ссылка на несколько ячеек с одним и тем же хэшем. Если хэш значения совпадают, то находятся по перебору из заданного диапозона.

Задача про расстановку 8 ферзей

