

РАЗРАБОТКА НА C++

Урок 7. Алгоритмы поиска

План

- Что такое алгоритмы поиска?
- Сферы применения алгоритмов поиска
- Какие есть популярные алгоритмы поиска?
- Линейный поиск
- Бинарный поиск
- Интерполяционный поиск
- Поиск с использованием хэш-таблиц
- Сравнение алгоритмов

Что такое алгоритмы поиска?

При разработке программ часто возникает необходимость в поиске определенных элементов в массивах или других структурах данных. Для эффективного и быстрого поиска используются алгоритмы поиска.

Иногда для упрощения поиска используется сортировка - упорядочивание элементов по определенному правилу. Она помогает найти нужные данные быстрее и сделать программы более эффективными.



Сферы применения алгоритмов поиска

Поисковые системы: Алгоритмы поиска используются для поиска веб-страниц, документов, изображений и другого контента в поисковых системах, таких как Google, Яндекс и Bing.

Базы данных: Алгоритмы поиска применяются для нахождения записей, информации или значений в базах данных. Это может быть полнотекстовый поиск, поиск по ключу или поиск по определенным критериям.

Анализ данных: Алгоритмы поиска используются для нахождения особенностей, закономерностей или схожих элементов в больших объемах данных. Это может включать поиск аномалий, кластеризацию или поиск паттернов.

Сферы применения алгоритмов поиска

Информационные системы: Алгоритмы поиска применяются для нахождения и извлечения информации из информационных систем, таких как электронные библиотеки, базы знаний или системы учета.

Мобильные приложения: Алгоритмы поиска используются в мобильных приложениях для поиска контактов, местоположений, продуктов или другой информации внутри приложения.

Робототехника: Алгоритмы поиска применяются в робототехнике для нахождения пути, объектов или сбора информации с помощью датчиков.

Игры: Алгоритмы поиска используются в компьютерных играх для принятия решений и определения оптимальных стратегий, поиска пути, нахождения противников и других игровых элементов.

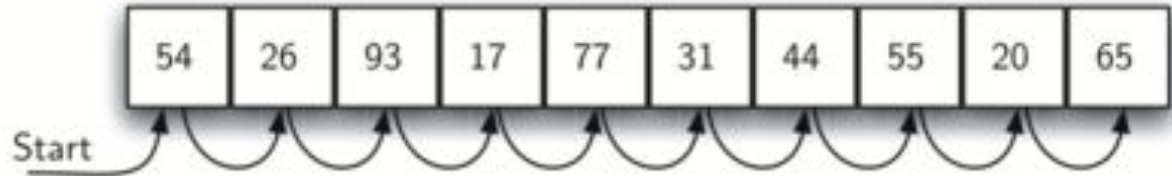
Какие есть популярные алгоритмы поиска?

- Линейный поиск
- Бинарный поиск
- Интерполяционный поиск
- Поиск с использованием хэш-таблиц



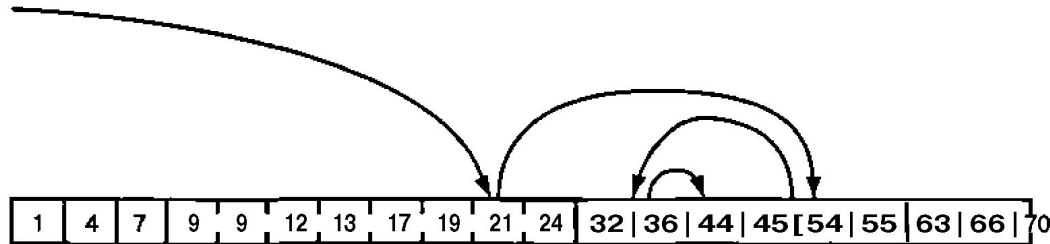
Линейный поиск

Линейный поиск - алгоритм проверяет каждый элемент в наборе данных пока не найдет нужный элемент или не дойдет до конца. Он начинает с первого элемента и сравнивает его с нужным значением. Если значения совпадают, поиск завершается.



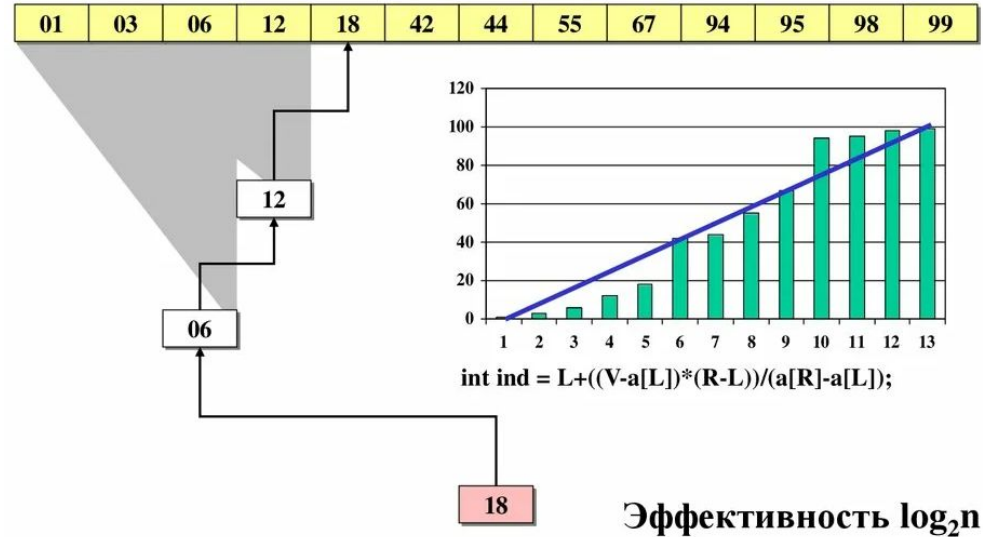
Бинарный поиск

Бинарный поиск - алгоритм поиска, который применяется к отсортированным данным. Суть его в делении массива пополам и сравнении нужного значения с элементом в середине. Если значение равно срединному элементу, поиск завершается. Если значение меньше срединного элемента, поиск продолжается в левой половине массива. Если значение больше срединного элемента, поиск продолжается в правой половине.



Интерполяционный поиск

Интерполяционный поиск - алгоритм поиска, который основан на интерполяции значений для нахождения более точного приближения нужного элемента. Интерполяционный поиск использует формулу для вычисления приближенного расположения искомого элемента в наборе данных. Он предполагает, что значения в наборе данных равномерно распределены.



Поиск с использованием хэш-таблиц

Поиск с использованием хеш-таблицы - алгоритм, который использует хеш-функцию для быстрого поиска элементов. Хеш-таблица представляет собой структуру данных, которая состоит из пар "ключ-значение". Хеш-функция преобразует ключ в уникальный хеш-код, который используется в качестве индекса для доступа к значению. При поиске элемента хеш-таблица вычисляет хеш-код для ключа и быстро находит соответствующее значение.

Пример хэш-таблицы

| | | | | | | | | | | |
|---|---|---|---|----|---|---|---|----|---|---|
| X | X | X | X | 37 | X | X | X | 52 | X | X |
|---|---|---|---|----|---|---|---|----|---|---|

Поиск элемента

16

$$16 \% 11 = 5$$

Не найден

| | | | | | | | | | | |
|---|---|---|---|----|---|---|---|----|---|---|
| X | X | X | X | 37 | X | X | X | 52 | X | X |
|---|---|---|---|----|---|---|---|----|---|---|

Поиск элемента

19

$$19 \% 11 = 8$$

Не найден

Сравнение алгоритмов. Линейный поиск

Преимущества: Простота реализации, работает для неотсортированных данных.

Недостатки: Медленный для больших объемов данных.



Сравнение алгоритмов. Бинарный поиск

Преимущества: Быстрый для отсортированных данных.

Недостатки: Требуется предварительной сортировки данных.

Сравнение алгоритмов. Интерполяционный поиск

Преимущества: Еще быстрее для равномерно распределенных данных.

Недостатки: Неэффективен для неравномерно распределенных данных.



Сравнение алгоритмов. Поиск с использованием хэш-таблицы

Преимущества: Очень быстрый для доступа по ключу.

Недостатки: Требуется дополнительная память для хранения хеш-таблицы.



Временная сложность

| Алгоритм поиска | Временная сложность в среднем случае | Временная сложность в худшем случае | Пространственная сложность |
|------------------------|--------------------------------------|-------------------------------------|----------------------------|
| Линейный поиск | $O(n)$ | $O(n)$ | $O(1)$ |
| Бинарный поиск | $O(\log n)$ | $O(\log n)$ | $O(1)$ |
| Интерполяционный поиск | $O(\log \log n)$ | $O(n)$ | $O(1)$ |
| Поиск в хэш-таблице | $O(1)$ | $O(n)$ | $O(n)$ |