

98. ИСПОЛЬЗОВАНИЕ АЛГОРИТМОВ ПОИСКА ДЛЯ РЕШЕНИЯ ЭКОНОМИЧЕСКИХ ЗАДАЧ

Насевич Д.Р. студентка группы 573901

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Кириенко Н.А. – кандидат технических наук, доцент

Аннотация. В данной работе исследуются теоретические аспекты и практическая значимость алгоритмов поиска в контексте обработки экономической информации. Рассматриваются критерии эффективности алгоритмов, их классификация и влияние на скорость принятия управленческих решений. Особое внимание уделено сравнительному анализу сложности методов и их реализации в современных аналитических системах.

Современная экономическая среда характеризуется переходом к модели «Data-Driven Management», где ключевым активом организации является информация. Постоянный рост объемов транзакционных данных, рыночных котировок и клиентских баз (Big Data) создает потребность в инструментах, способных обеспечивать мгновенный доступ к нужным сведениям. Алгоритмы поиска являются фундаментом вычислительной логики, позволяя не просто хранить, а эффективно использовать информационные массивы для прикладного анализа. Без оптимизированных механизмов поиска экономический анализ сталкивается с проблемой «информационного шума», когда время на извлечение данных превышает время их полезного использования.

Методология оценки эффективности.

Выбор алгоритма для решения конкретной экономической задачи не может быть произвольным. Он основывается на математической оценке эффективности, которая выражается через асимптотическую сложность — Big O notation.

Временная сложность — отражает зависимость времени выполнения алгоритма от объема входных данных (n). В экономике это критично при расчете высокочастотных операций на бирже или обработке миллионов банковских проводок.

Пространственная сложность — определяет объем оперативной памяти, необходимый для работы алгоритма. При работе с облачными хранилищами и распределенными базами данных минимизация потребления памяти позволяет существенно снизить инфраструктурные издержки [1].

Согласно классификации, алгоритмы с константной $O(1)$ и логарифмической сложностью $O(\log n)$ считаются наиболее эффективными для крупномасштабных систем [2].

Классификация и сравнительный анализ поисковых механизмов.

В практике финансового мониторинга и бизнес-аналитики выделяют три основных алгоритма:

Линейный поиск ($O(n)$): Алгоритм последовательного перебора элементов. Его применение в экономике ограничено малыми выборками (например, поиск в рамках одного чека или небольшого списка контрагентов). Главный недостаток — прямо пропорциональный рост затрат времени при увеличении объема данных. Если массив данных увеличивается в k раз, время поиска также возрастает в k раз, что делает его непригодным для анализа больших рынков.

Иерархические методы: Бинарный поиск ($O(\log n)$). Данный метод является «золотым стандартом» для работы с упорядоченными реестрами. Принцип «разделяй и властвуй» позволяет сокращать область поиска вдвое на каждом шаге. Для экономики это означает, что среди 1 000 000 записей нужная транзакция будет найдена максимум за 20 итераций [3]. Обязательным условием здесь выступает предварительная сортировка данных, что требует дополнительных вычислительных мощностей на этапе подготовки.

Интерполяционный поиск и ассоциативные массивы. Для данных с известным распределением (например, инвентарные номера или даты) используется интерполяционный поиск, который рассчитывает вероятное местоположение элемента. В современных СУБД для достижения максимальной скорости ($O(1)$) применяются хэш-таблицы. Хэширование позволяет мгновенно получать данные о клиенте по его уникальному идентификатору (ID), минуя этап сравнения всех элементов [4].

Технологическая реализация и оптимизация в среде Big Data.

Реализация алгоритмов в современных экономических отделах редко происходит «с нуля». Основной упор делается на использование высокопроизводительных библиотек.

Pandas и NumPy (Python). Данные библиотеки позволяют проводить векторизованный поиск, который выполняется на уровне низкоуровневых инструкций процессора, что в десятки раз быстрее стандартных циклов.

Индексация баз данных. Создание индексов в SQL-базах данных позволяет превратить медленный линейный поиск в высокоскоростной логарифмический, что является стандартом для банковских систем.

Параллельные вычисления. Распределение поискового запроса по нескольким ядрам или серверам позволяет обрабатывать терабайты данных в реальном времени, что необходимо для систем обнаружения мошенничества.

Практические рекомендации по выбору алгоритмов.

Для оптимизации экономических вычислений рекомендуется следовать алгоритму выбора.

Для малых данных (<1000 записей) допустимо использование линейного поиска ввиду низких затрат на разработку.

Для статичных больших данных целесообразна разовая сортировка с последующим многократным использованием бинарного поиска.

Для динамических потоков данных применяются хэш-индексации для мгновенного доступа без необходимости постоянной пересортировки.

Список использованных источников:

1. Зайнуллина А.И., Куценко С. М. Применение алгоритмов сортировки и поиска в экономическом анализе данных / А.И. Зайнуллина, С.М. Куценко. – Казанский государственный энергетический университет, Казань, 2025. — 8 с
2. Кайда Н. Что такое «О» большое: объясняем на простых примерах / Н. Кайдар. / Proglib, 2024. — URL: <https://proglib.io/p/chto-takoe-o-bolshoe-2024-04-27> – Дата обращения: 24.03.2026.
3. Это база. Алгоритмы сортировки и поиска для начинающих / Habr / Selectel, 2024. — URL: <https://habr.com/ru/companies/selectel/articles/851206/>. – Дата обращения: 24.03.2026.
4. 6 алгоритмов поиска на Java: от простого к сложному / Proglib: образовательный портал, 2024. – URL: <https://proglib.io/p/6-search-algorithms-java>. – Дата доступа: 24.03.2026.