МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное

учреждение высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ОТЧЁТ**

по лабораторной работе №5

по дисциплине «Алгоритмы и структуры данных»

**Преподаватель:** Шишкин Вадим Викторинович

**Группа:** ИСТбд-11

**Студент:** Саранцев Семён Александрович

**г. Ульяновск**

**2024**

**Лабораторная работа №5**

**Цель работы:** Исследование и сравнение времени выполнения трёх различных методов вычисления последовательности F(n): рекурсивного, итерационного и динамического (с использованием кэширования).

**Решение:**

В ходе выполнения лабораторной работы были созданы три функции для вычисления значений последовательности F(n):

1. **Рекурсивная функция (recursive\_factorial) -** вычисляет F(n) через рекурсивные вызовы самой себя.
2. **Итерационная функция (iterative\_factorial) -** вычисляет F(n), используя итерационный подход, который заменяет рекурсию циклом.
3. **Динамическая функция (dynamic\_factorial) -** использует динамическое программирование с кэшированием промежуточных результатов для ускорения вычислений.

Каждый метод был оценен на основе времени, затрачиваемого на вычисление F(n) для заданного значения n. Для измерения времени использовалась функция **score\_time**, которая принимает целевую функцию и значение n, очищает кэши и вычисляет среднее время выполнения функции с помощью модуля **timeit**.

**Расчеты:**

С применением модуля **timeit** для точного измерения времени выполнения и библиотеки **matplotlib** для графического представления данных, были получены следующие результаты выполнения разработанных функций:

* **Рекурсивная функция**: Замечено, что время выполнения этого метода увеличивается экспоненциально по мере увеличения **n**. Это объясняется большим количеством повторных вызовов функции для одних и тех же значений.
* **Итеративная функция**: Время выполнения этого метода растет линейно с увеличением **n**. Итеративный подход оказался более предсказуемым и стабильным, поскольку каждое значение вычисляется один раз с постоянным количеством операций на каждом шаге.
* **Функция с использованием динамического программирования**: Благодаря использованию мемоизации, то есть сохранению уже вычисленных значений, динамический метод показал наиболее эффективное время выполнения. Сложность вычислений существенно снижается за счет исключения необходимости повторного вычисления уже известных значений.

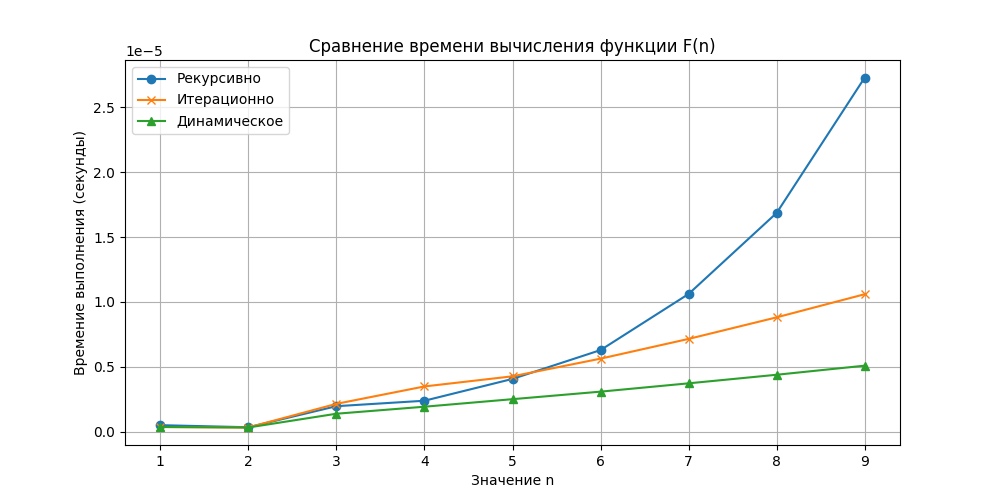
**Выводы по лабораторной работе:**

На основе проведённых измерений и анализа графика можно сделать следующие выводы:

* Рекурсивный метод оказался наименее эффективным из-за значительного количества повторных вычислений и большой глубины рекурсивных вызовов.
* Итеративный метод показал улучшенные результаты по сравнению с рекурсивным, так как исключает избыточные вычисления за счёт последовательного выполнения операций.
* Динамический метод продемонстрировал наилучшие показатели времени выполнения, подтверждая эффективность кэширования результатов для ускорения вычислений, особенно при больших n.

Таким образом, визуализация результатов позволяет наглядно сравнить различные подходы и подтверждает предположение о том, что методы с кэшированием и оптимизацией вычислений могут значительно повысить производительность, что особенно важно при работе с ресурсоёмкими задачами.

**Графический отчет к лабораторной работе №5**



**Табличный отчет к лабораторной работе №5**

