



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени
Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 7
по дисциплине
«Функциональное и логическое
программирование»

Тема Рекурсивные выражения

Студент Александров Э.И.

Группа ИУ7-53БВ

Преподаватель Строганов Ю.В.

Москва, 2024

Содержание

ВВЕДЕНИЕ	4
1 Аналитическая часть	5
2 Конструкторская часть	6
3 Технологическая часть	7
ЗАКЛЮЧЕНИЕ	9
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	10
Приложение А	11

ВВЕДЕНИЕ

Цель данной работы заключается в реализации вычисления элементов ряда Фибоначчи и факториала числа с использованием хвостовой и нехвостовой рекурсии. Задачи, поставленные в ходе работы, включают изучение принципов рекурсивного программирования, сравнение эффективности различных подходов и анализ полученных результатов.

1 Аналитическая часть

Рекурсия — это метод, при котором функция вызывает саму себя для решения подзадач. Существует два основных типа рекурсии: нехвостовая и хвостовая. Нехвостовая рекурсия требует сохранения состояния вызова функции, что может привести к увеличению использования памяти и времени выполнения. Хвостовая рекурсия, в свою очередь, позволяет компилятору оптимизировать вызовы функций, что делает её более эффективной с точки зрения использования ресурсов. В данной работе мы рассмотрим оба подхода на примере вычисления чисел Фибоначчи и факториала.

Вывод

В аналитической части мы изучили основные принципы рекурсии, а также различия между хвостовой и нехвостовой рекурсией.

2 Конструкторская часть

В данной части работы были разработаны алгоритмы для вычисления чисел Фибоначчи и факториала с использованием обоих типов рекурсии. Для реализации алгоритмов использовался язык программирования Lisp. Локальные функции были использованы для реализации хвостовой рекурсии, что позволило оптимизировать выполнение.

Вывод

В конструкторской части мы разработали алгоритмы для вычисления чисел Фибоначчи и факториала, реализовав их как с хвостовой, так и с нехвостовой рекурсией.

3 Технологическая часть

Реализация была выполнена на языке программирования Lisp. Для написания и тестирования кода использовалась среда разработки VS Code с установленным пакетом Common Lisp, а также компилятор Steal Bank Common Lisp. Код для вычисления чисел Фибоначчи и факториала представлен ниже:

Коды алгоритмов:

Листинг 3.1 — Фибоначчи нехвостовая рекурсия

```
(defun fibonacci-non-tail (n)
  (if (<= n 1)
      n
      (+ (fibonacci-non-tail (- n 1))
         (fibonacci-non-tail (- n 2))))))
```

Листинг 3.2 — Фибоначчи хвостовая рекурсия

```
(defun fibonacci-tail (n)
  (labels ((fib-helper (a b count) ;; local fun
            (if (= count 0)
                a
                (fib-helper b (+ a b) (- count 1)))))
    (fib-helper 0 1 n)))
```

Листинг 3.3 — Факториал нехвостовая рекурсия

```
(defun factorial-non-tail (n)
  (if (<= n 1)
      1
      (* n (factorial-non-tail (- n 1)))))
```

Листинг 3.4 — Факториал хвостовая рекурсия

```
(defun factorial-tail (n)
  (labels ((fact-helper (n acc)
            (if (<= n 1)
                acc
                (fact-helper (- n 1) (* n acc)))))
    (fact-helper n 1)))
```

Тесты:

- 1) (fibonacci-non-tail 10) - Вернет 55
- 2) (factorial-non-tail 5) - Вернет 120
- 3) (fibonacci-tail 10) - Вернет 55

4) (factorial-tail 5) - Вернет 120

Тестирование алгоритмов показало, что все тесты пройдены успешно, и результаты вычислений соответствуют ожидаемым значениям.

Вывод

В технологической части мы реализовали алгоритмы на языке Lisp, протестировали их и подтвердили корректность работы.

ЗАКЛЮЧЕНИЕ

В ходе работы была поставлена цель реализовать вычисление элементов ряда Фибоначчи и факториала числа с использованием хвостовой и нехвостовой рекурсии. Все задачи были успешно выполнены, и полученные результаты подтвердили эффективность хвостовой рекурсии по сравнению с нехвостовой.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Пол Грэм, ANSI Common Lisp - СПб.: Символ-Плюс, 2012. - 448 с.
2. Сайт stackoverflow.com — [<https://stackoverflow.com/questions/33923/what-is-tail-recursion>].
3. Сайт geeksforgeeks.org — [<https://www.geeksforgeeks.org/recursion-in-lisp/>]
4. Сайт tailrecursion.com — [<https://tailrecursion.com/alan/Lisp/CommonLispIteration.html>]

Приложение А

Код программы:

Листинг 3.5 — Фибоначчи нехвостовая рекурсия

```
(defun fibonacci-non-tail (n)
  (if (<= n 1)
      n
      (+ (fibonacci-non-tail (- n 1))
         (fibonacci-non-tail (- n 2))))))
```

Листинг 3.6 — Фибоначчи хвостовая рекурсия

```
(defun fibonacci-tail (n)
  (labels ((fib-helper (a b count) ;; local fun
    (if (= count 0)
        a
        (fib-helper b (+ a b) (- count 1)))))
    (fib-helper 0 1 n)))
```

Листинг 3.7 — Факториал нехвостовая рекурсия

```
(defun factorial-non-tail (n)
  (if (<= n 1)
      1
      (* n (factorial-non-tail (- n 1)))))
```

Листинг 3.8 — Факториал хвостовая рекурсия

```
(defun factorial-tail (n)
  (labels ((fact-helper (n acc)
    (if (<= n 1)
        acc
        (fact-helper (- n 1) (* n acc)))))
    (fact-helper n 1)))
```