

Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики

Факультет информационных технологий и программирования
Кафедра компьютерных технологий

Рыбак Андрей Викторович

**Представление структур данных индуктивными
семействами и доказательства их свойств**

Научный руководитель: ассистент кафедры ТП Я. М. Малаховски
Санкт-Петербург

2014

Содержание

Введение	4
Глава 1. Обзор предметной области	5
1.1 Структуры данных	5
1.1.1 Функциональные структуры данных	5
1.2 Индуктивные семейства и зависимые типы	5
1.3 Agda	6
1.4 Выводы по главе 1	7
Глава 2. Описание реализованной структуры данных	8
2.1 Постановка задачи	8
2.2 Структура данных «двоичная куча»	8
2.2.1 Заполнение дерева	9
2.3 Тип данных для двоичной кучи	9
2.4 Вставка элементов	9
2.4.1 Вставка элементов в заполненное дерево	9
2.4.2 Вставка элементов в незаполненное дерево	9
2.5 todo3	9
2.5.1 subsection	9
2.5.2	9
2.6	9
2.7 Выводы по главе 2	9
Заключение	10
Список литературы	11

Введение

Структуры данных используются в программировании повсеместно для упрощения хранения и обработки данных. Свойства структур данных происходят из инвариантов, которые эта структура данных соблюдает.

Практика показывает, что тривиальные структуры и их инварианты данных хорошо выражаются в форме индуктивных семейств. Мы хотим узнать насколько хорошо эта практика работает и для более сложных структур.

В данной работе рассматривается представление в форме индуктивных семейств структуры данных приоритетная очередь типа «двоичная куча».

Глава 1. Обзор предметной области

В программировании структуры данных позволяют упростить хранение и обработку множество однотипных и/или логически связанных данных. Задача структур данных — облегчить написание программ для программистов и ускорить обработку данных.

1.1. СТРУКТУРЫ ДАННЫХ

Структуры данных используются в программировании для абстрагирования обработки связанных и однородных данных.

Часто используемые структуры данных включаются в стандартные библиотеки языков программирования.

1.1.1. Функциональные структуры данных

Главное отличие функциональных структур данных от императивных [1] заключается в неиспользовании разрушающих обновлений (то есть присваиваний). При обновлении структуры данных все измененные части создаются заново.

1.2. ИНДУКТИВНЫЕ СЕМЕЙСТВА И ЗАВИСИМЫЕ ТИПЫ

Определение 1.1. Индуктивное семейство [2]— это семейство типов данных, которые могут зависеть от других типов и значений.

Одной из областей применения индуктивных семейств являются системы интерактивного доказательства теорем.

Индуктивные семейства позволяют формализовать математические структуры, кодируя утверждения о структурах в них самих, тем самым перенося сложность из доказательств в определения.

1.3. AGDA

Agda [3] — чистый функциональный язык программирования с зависимыми типами.

В Agda есть поддержка модулей.

```
module AgdaDescription where
```

В коде на Agda широко используются символы Unicode. Пример — определение типа натуральных чисел.

```
data ℕ : Set where
  zero : ℕ
  succ  : ℕ → ℕ
```

В Agda функции можно определять как `mixfix` операторы. Пример — сложение натуральных чисел:

```
_+_ : ℕ → ℕ → ℕ
zero + b = b
succ a + b = succ (a + b)
```

Символы подчеркивания обозначают места для аргументов.

Зависимые типы позволяют определять типы, зависящие (или индексированные) от значений других типов. Пример — список, индексированный длиной:

```
data Vec (A : Set) : ℕ → Set where
  nil  : Vec A zero
  cons : ∀ {n} → A → Vec A n → Vec A (succ n)
```

В фигурные скобки заключаются неявные аргументы.

Такое определение позволяет нам описать функцию `head` для такого списка, которая не может бросить исключение:

$\text{head} : \forall \{A\} \{n\} \rightarrow \text{Vec } A (\text{succ } n) \rightarrow A$

У аргумента функции `head` тип $\text{Vec } A (\text{succ } n)$, то есть вектор, в котором есть хотя бы один элемент. Это позволяет произвести сопоставление с образцом только по конструктору `cons`:

$\text{head } (\text{cons } a \text{ as}) = a$

1.4. ВЫВОоды по главе 1

Рассмотрены некоторые существующие подходы к Описаны различные Кратко описана Кратко описаны особенности языка программирования *Agda*.

Глава 2. Описание реализованной структуры данных

В данной главе описывается разработанная функциональная структура данных приоритетная очередь типа «двоичная куча».

2.1. ПОСТАНОВКА ЗАДАЧИ

Целью данной работы является разработка типов данных для представления структуры данных и инвариантов.

Требования к данной работе:

- Разработать типы данных для представления структуры данных
- Реализовать функции по работе со структурой данных
- Используя разработанные типы данных доказать выполнение инвариантов.

2.2. СТРУКТУРА ДАННЫХ «ДВОИЧНАЯ КУЧА»

Определение 2.1. Двоичная куча или пирамида [4] — такое двоичное подвешенное дерево, для которого выполнены следующие три условия:

- Значение в любой вершине не больше (если куча для минимума), чем значения её потомков.
- На i -ом слое 2^i вершин, кроме последнего. Слои нумеруются с нуля.
- Последний слой заполнен слева направо (как показано на рисунке 2.1).

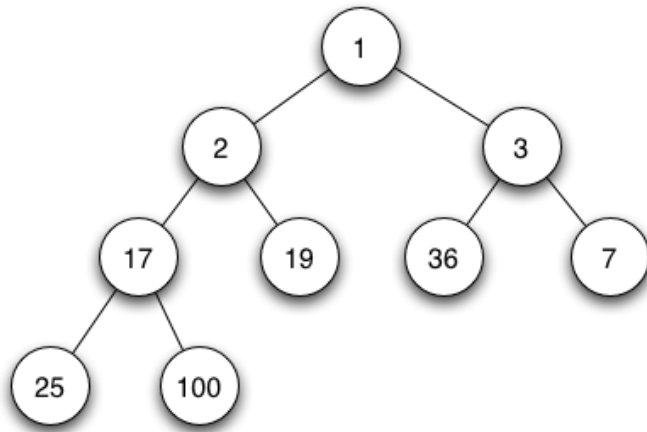


Рис. 2.1: Пример заполненной кучи для минимума

2.2.1. Заполнение дерева

2.3. Тип данных для двоичной кучи

2.4. ВСТАВКА ЭЛЕМЕНТОВ

2.4.1. Вставка элементов в заполненное дерево

2.4.2. Вставка элементов в незаполненное дерево

2.5. TODO3

2.5.1. subsection

2.5.2.

2.6.

2.7. ВЫВОДЫ ПО ГЛАВЕ 2

Разработаны типы данных для представления структуры данных двоичная куча.

Заключение

В данной работе реализована структура данных «двоичная куча». Разработаны типы данных для представления структуры данных и инвариантов индуктивными семействами. Реализованы функции для работы со структурой данных. Было доказано соблюдение инвариантов с использованием вспомогательных типов данных.

Таким образом, данная работа удовлетворяет поставленным требованиям.

Список литературы

1. *Okasaki C.* Purely functional data structures. Cambridge University Press, 1999. С. I–X, 1–220. ISBN: 978-0-521-66350-2.
2. *Dybjer P.* Inductive Families // Formal Asp. Comput. 1994. №4. С. 440–465.
3. Agda language. <http://wiki.portal.chalmers.se/agda/pmwiki.php>.
4. *Cormen T. H., Leiserson C. E., Rivest R. L., Stein C.* Introduction to Algorithms, Second Edition. The MIT Press и McGraw-Hill Book Company, 2001. ISBN: 0-262-03293-7, 0-07-013151-1.