



Доказательство сортировки списков алгоритмом Хоара

Автор: Маллабаев Азамат Нурмухаматович, 143 группа
Научный руководитель: ст.пр. С.В. Григорьев

Санкт-Петербургский государственный университет

2015

Разработан Чарльзом Хоаром во время работы в МГУ в 1960 году
Эффективность порядка от $O(n \log n)$ до $O(n^2)$

Алгоритм сортировки Хоара

- ❶ Взять опорный элемент (берем первый элемент списка)
- ❷ Разбить список на 3 части, где элементы
 - ❶ Меньше опорного
 - ❷ Равны опорному
 - ❸ Больше опорного
- ❸ Отсортировать отдельно 1ую и 3ую часть
- ❹ Объединить все 3 части

Алгоритмы сортировки

- Сортировка пузырьком
- Шейкерная сортировка
- Сортировка слиянием

Инструмент реализации и доказательства - F*

Целью работы является доказательство сортировки Хоара для списков

Задачи

- 1 Реализовать алгоритм для сортировки списка чисел
- 2 Провести доказательство корректности алгоритма

Реализация алгоритма

Определение

```
module HoarSort
type List 'a =
- | Empty
- | Smth: hd:'a -> tl:List 'a -> List 'a
val part: int -> List int -> Tot (List int * List int * List int)
val merge: List int * List int * List int -> Tot (List int)
val qsort: List int -> Tot (List int)
let rec qsort lst =
- match lst with
- | Empty -> Empty
- | Smth h t ->
- - let al, bl, sl = part h lst
- - merge (qsort al) bl (qsort sl)
```

Доказательство корректности алгоритма

Доказательство

```
module Proof
open HoarSort
val sorted: MyList int -> Tot bool
let sorted lst =
- match lst with
- | Empty -> true
- | Smth a Empty -> true
- | Smth a (Smth b t) ->
- - if a > b
- - then false
- - else sorted (Smth b t)
val sortedHoarSortResult: MyList 'a -> Tot unit
let sortedHoarSortResult lst =
- assert(sorted (qsort lst))
```

Выполнены задачи

- Реализована сортировка Хоара для списков чисел
- Проведено доказательство корректности алгоритма

В ходе работы выявлены проблемы инструмента доказательства

- Неустоявшийся синтаксис
- Недоработанный верификатор
- Неустоявшийся интерфейс