

# Представление структур данных индуктивными семействами и доказательства их свойств

НИУ ИТМО

Май 2014

*Индуктивное семейство* — это семейство типов данных, которые могут зависеть от других типов и значений.

```
module AgdaPresentation where
```

```
data  $\mathbb{N}$  : Set where
```

```
  zero :  $\mathbb{N}$ 
```

```
  succ :  $\mathbb{N} \rightarrow \mathbb{N}$ 
```

```
_+_ :  $\mathbb{N} \rightarrow \mathbb{N} \rightarrow \mathbb{N}$ 
```

```
zero + b = b
```

```
succ a + b = succ (a + b)
```

```
data Vec (A : Set) :  $\mathbb{N} \rightarrow$  Set where
```

```
  nil : Vec A zero
```

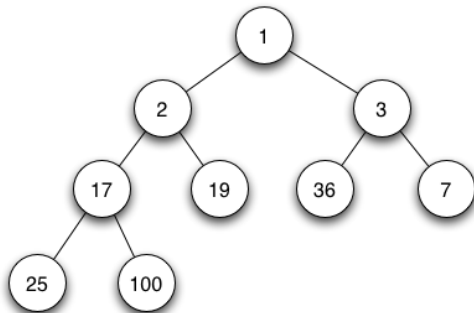
```
  cons :  $\forall \{n\} \rightarrow A \rightarrow \text{Vec } A \ n \rightarrow \text{Vec } A \ (\text{succ } n)$ 
```

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

```
head (cons a as) = a
```

- двоичное дерево
- заполняется слева
- значение в узле  $\leq$  значений в корнях поддеревьев

# Пример



```
data TreeState : Set where
  full almost : TreeState
```

```
data Tree : (h : ℕ) → TreeState → Set where
  et : Tree zero full - Пустое дерево
  nf : ∀ {n} → (a : Tree n full) → (b : Tree n full)
    → Tree (succ n) full - Полное дерево
  nd : ∀ {n} → (a : Tree (succ n) full) → (b : Tree n full)
    → Tree (succ (succ n)) almost - Полные поддеревья разной высот
  nl : ∀ {n} → (a : Tree (succ n) almost) → (b : Tree n full)
    → Tree (succ (succ n)) almost - Правое поддерево - полное
  nr : ∀ {n} → (a : Tree n full) → (b : Tree n almost)
    → Tree (succ n) almost - Левое поддерево - полное
```