# LENE TRAJNE PODATKOVNE STRUKTURE

Stroški operacij:

- Lastni (vse kar se zgodi takoj)
- Skupni (nastavljeni izračuni, memoizacija)

Oznake:

- $S(v)$ ... Operacija nastavi (suspension/okraski) izračun (se zadolži)
- $r(v)$ ... Operacija požre nastavljen izračun (realizirani stroški) kot vsoto po uniji operacij
- $a(v)$ ... Del amortizirane cene prispeva k odplačilu dolga

Upoštevamo:

△ Odplačujemo samo najete izračune (ne varčujemo)

△ Izračun odplačamo, preden ga poženemo

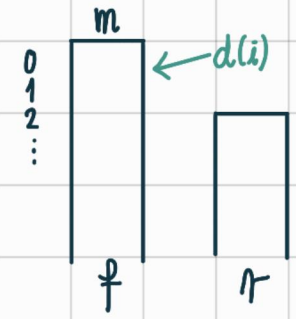
△ Vsaka prihodnost poskrbi za plačilo svojih operacij

---

# VRSTE

$Q_0$

$Q_i$




$f$ ++ (reverse $r$)

$$D(i) = \sum_{k=1}^{i} d(k)$$

vstavi/snoc odplača 1
rep/tail odplača 2

$D(i) \leq \min(2i, |f|-|r|)$

| | $f$ | $d(i)$ | $D(i)$ | $r$ |
|---|---|---|---|---|
| | ~~0~~ | 0 | 0 | |
| 0 | 1 | 0 | 0 | 4 |
| 1 | 2 | 0 | 0 | 3 |

$$d(i) = \begin{cases} 1 & ; \ i < m \\ m+1 & ; \ i = m \\ 0 & ; \ i > 0 \end{cases}$$

Vstavljanje brez rotacije:

- $|r|$ ++
- $|f|$, i ostaneta enaka
- $(|f|-|r|)$ --

⇒ D(i) kvečjemu manjši za 1

Ok, če odplačamo 1

Rep brez rotacije:

- $|f|$ --
- $(|f|-|r|)$ --

⇒ Ok, če odplačamo 2

Vstavljanje z rotacijo (ko $|f|<|r|$):

⇒ Vstavljamo in naredimo rotacijo

---

22.5.

# RANDOM ACCESS LIST (RLIST)

Primer: [  ]

---

ustvari :: int × α → α RList

... naredi seznam z n kopijami elementa a v $O(\log n)$

$n = 2^k - 1$ : [  ]

sicer: po potrebi nimamo vseh povezav

Primer: $n=2$: [0, a  ]

```
fun create(n,x):
   let fun build(0, tree) = []
         | build(n, tree) =
              (if n mod 2 = 1 then ONE tree else ZERO)
               :: build(n div 2, link(tree, tree))
   in build(n, LEFT x)
end
```

---

# SPARSE RANDOM ACCESS LIST

```
fun cons(t,[]) = [t]
  | cons(t, ts as t'::ts') =
      if sizet < sizet' then t :: ts
      else cons(link(t,t'), ts')

fun uncons [] = raise "EMPTY"
  | uncons (LEAF x::ts) = (x, ts)
  | uncons (NODE(_,t1,t2)::ts) = uncons t1::t2::ts
```

---

# BREZNIČELNA REPREZENTACIJA

$10 \longrightarrow \frac{10-2}{2} = 4 \longrightarrow \frac{4-2}{2} = 1 \longrightarrow \frac{1-1}{2} = 0$

↓ soda ↓ soda ↓ liha

TWO TWO ONE

$5 \longrightarrow \frac{5-1}{2} = 2 \longrightarrow \frac{2-2}{2} = 0$

↓ liha ↓ soda

ONE TWO

Odštevanje števil v brezničelni reprezentaciji:

```
fun dec [] = raise "FAIL"
  | dec [ONE] = []
  | dec [TWO] = [ONE]
  | dec [ONE :: ds] = TWO :: dec ds
  | dec [TWO :: ds] = ONE :: ds
```

# POŠEVNA DVOJIŠKA ŠTEVILA

$$n = \sum_{i=0}^{\dots} d_i (2^{i+1} - 1), \quad d_i \in \{0,1,2\}$$

$$d_i = 2 \Rightarrow \forall j < i : d_j = 0$$

Primer: $1120 = 0 \cdot 1 + 2 \cdot 3 + \dots = [3,3,7,15]$

$$\text{naslednik}(x :: x :: xs) = (2x+1) :: xs$$
$$\text{naslednik}(x :: y :: xs) = 1 :: x :: y :: xs \quad (x \neq y)$$

$$\text{predhodnik}(1 :: xs) = xs$$
$$\text{predhodnik}((2x+1) :: xs) = x :: x :: xs$$

---

Dokaži: Vsako naravno število ima enoličen zapis

Indukcija po $m$, kjer je $m \geq n$ ...

$$\sum_{i=0}^{m} d_i (2^{i+1} - 1) = \sum_{i=0}^{n} b_i (2^{i+1} - 1)$$

$m = 0$:

Očitno ☺

$m > 0$:

$$\sum_{i=0}^{m} d_i(2^{i+1}-1) = \sum_{i=0}^{n} b_i(2^{i+1}-1) \leq 1\ldots12 = 2 + \sum_{i=1}^{n}(2^{i+1}-1) = 2^{n+2}-n-2$$

$$\wedge$$

$$2^{n+1}-1$$

Če $m > n$, je $2^{m+1} \geq 2^{n+1}$.

Potem je $2^{m+1}-1 > 2^{n+2}-n-2$. $\longleftrightarrow$

Torej $m = n$.

Recimo $d_n \neq b_n$.

BŠS: $d_n = 1$, $b_n = 2$

$\Rightarrow$ Števili sta $2 \cdot (2^{n+1}-1) = \sum_{i=0}^{n} d_i(2^{i+1}-1) \leq 2^{n+2}-n-2$

$2^{n+2}-2$

$\Rightarrow -2 \leq -n-2$

 (za $n \neq 0$)

Sledi po indukciji.