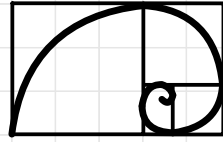


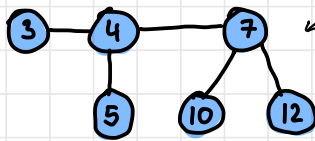
Lopce Fibonacciego /

Kopce Fibonacciego



Kopiec Fibonacciego jest podobny do lewej implementacji kopca drzewianowego, jednakże drzewa w tej strukturze niekoniecznie muszą być drzewianowe.

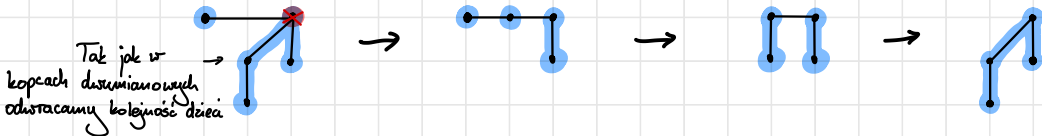
Przykład



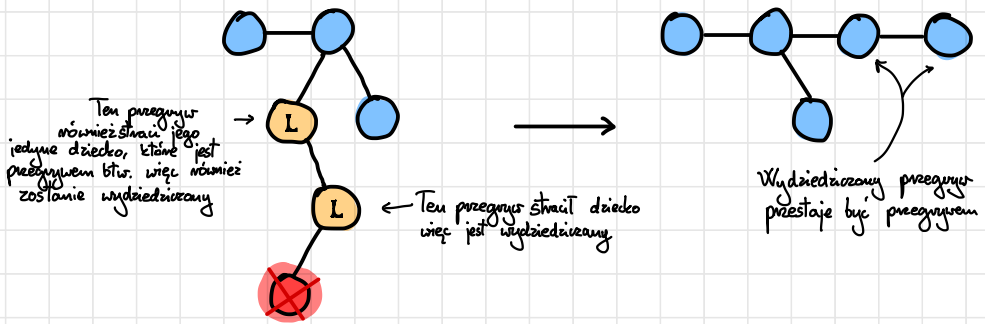
Drzewa w kopcu Fibonacciego nie muszą być drzewianowe

Cechy kopca Fibonacciego

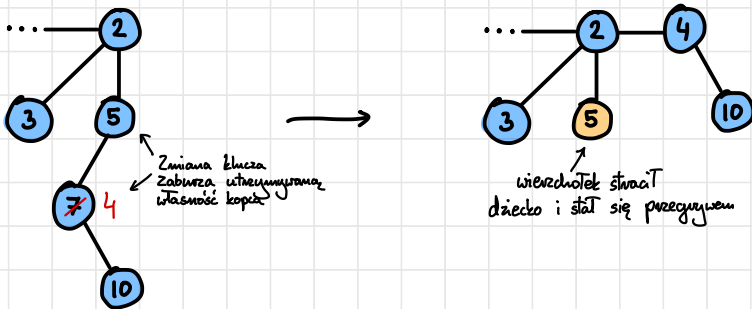
- Dodawanie elementu jest po prostu dodawaniem nowego wierzchołka do listy drzew.
- **Delete min** - usuwamy wierzchołek, dzieci są teraz nowymi drzewami, a następnie scalamy **wszystkie** drzewa o tym samym stopniu.



- Wierzchołek, którego opuściło co najmniej jedno dziecko staje się przegrupem **L** (looser).
- Wierzchołek, który jest przegrupem **L** stracił następne dziecko - jest wydzielony (zrzuć) przeniesiony do listy wężanej)



- Korzeń nie może być przegrymem
- **Decrease key** - przenosimy ten węzeł wraz z jego dziećmi do listy wiązanej jako nowe drzewo jeśli jest mniejszy niż jego rodzic:



Implementacja w Dijkstra

W algorytmie Dijkstra wykonujemy $|E|$ razy **decrease key** na kolejce priorytetowej, czyli sprawdzamy każdą krawędź, która może zmniejszyć odlegość do jakiegoś wierzchołka.

Dodatkowo $|V|$ razy wykonujemy **delete-min**, aby usunąć następną najbliższą wierzchołek po sprawdzeniu wszystkich niesprawdzonych dotychczas krawędzi z odwiedzonych przez nas wierzchołków.

```
procedure Dijkstra
```

```
   $X \leftarrow \{s\}$ 
```

```
   $D(s) \leftarrow 0$ 
```

```
  for each  $u \in V \setminus \{s\}$  do  $D(u) \leftarrow l(s, u)$ 
```

```
  while  $X \neq V$  do
```

```
    Niech  $u \in V \setminus X$  o minimalnej wartości  $D(u)$ 
```

```
     $X \leftarrow X \cup \{u\}$ 
```

```
    for each  $\langle u, v \rangle \in E$  takiej, że  $v \in V \setminus X$  do
```

```
       $D(v) \leftarrow \min(D(v), D(u) + l(u, v))$ 
```

Dlatego kopce Fibonacciego są tutaj tak wartościowe, ponieważ **decrease-key** wykonujemy w $O(1)$, a **delete-min** zajmie nam $O(\log n)$.

Zatem złożoność Dijkstra wynosi

$$O(|E| + |V| \log |V|)$$