Karlsruher Institut für Technologie

Übung 1 – Algorithmen II

Michael Axtmann – *michael.axtmann@kit.edu*

http://algo2.iti.kit.edu/AlgorithmenII_WS16.php

Institut für Theoretische Informatik - Algorithmik II

Organisatorisches-Übungsbetrieb

- Diskussionsforum
  - Nutzt das Forum
  - Wir raten, was Ihr nicht verstanden haben könntet. . .
    . . . Ihr wisst, was Ihr nicht verstanden habt
  - Themen aus Forum können für die Übung aufgegriffen werden

- Übungsblätter
  - Bearbeitung zwar freiwillig . . .
  - . . . aber sehr gute Vorbereitung auf die Klausur
  - Erstes Blatt Dienstag nächste Woche, noch viele kleine Aufgaben
  - Fast alle klausurtypisch

Organisatorisches

- Klausur: Mittwoch, den 22.02.2017, von 14:30 bis 16:30 Uhr
- Erasmus
  - Prüfungserlaubnis
    - Erasmus Koordination
    - Frau Gheta (Raum 124, (Geb. 50.34)
  - Fristgerecht im Sekretariat abgeben
    - Frau Blancani (Raum 218, (Geb. 50.34)
- Youtube-Stream der Vorlesung

Amortisierte Analyse

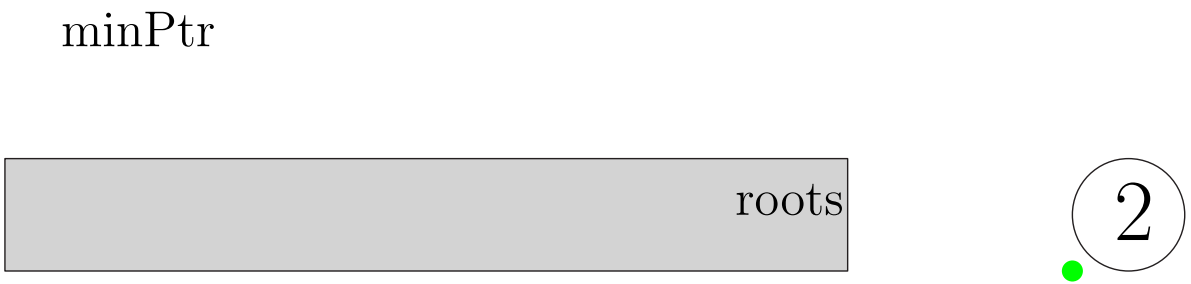
- Woher kommt die Rechenzeit?
  - **Bestendes Token**: Rechenzeit schon bezahlt
  - **Neues Token**: zählt zur amortisierten Laufzeit der Operation
- Kann ein Token ein anderes Token erstellen?

Legende

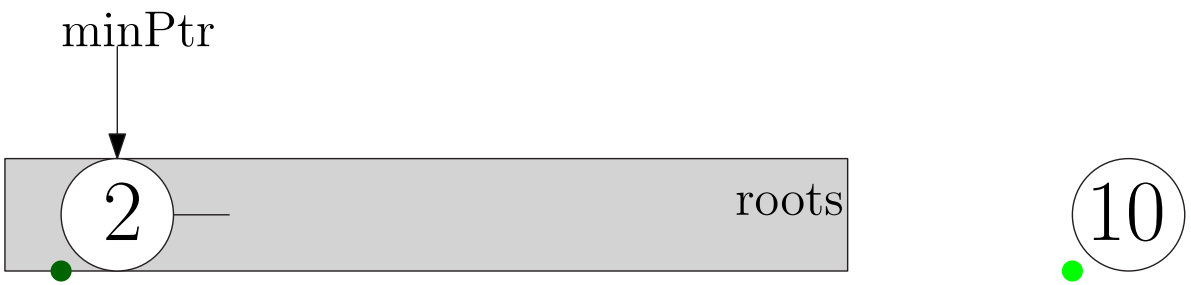
Karlsruher Institut für Technologie

- ✕ Markierung - Kennzeichnet Knoten, die ein Kind verloren haben
- ✕✕ Doppelte Markierung - Temporäre Markierung. Muss sofort aufgelöst werden
- ● Neues, unbenutztes Token
- ● InsertTree-Token - kann genau 1 InsertTree Operation bezahlen
- ● Bezahltoken - wird zum Bezahlen genutzt
- ● Union-Token - zahlt für 1 Union Operation; zusätzlich einen InsertTree-Token erstellen
- ● Mark Token - Zahlt Cut und Markierung des Parents
- ● ● Aktive Knoten
- ● Knoten zum Löschen oder Cutten

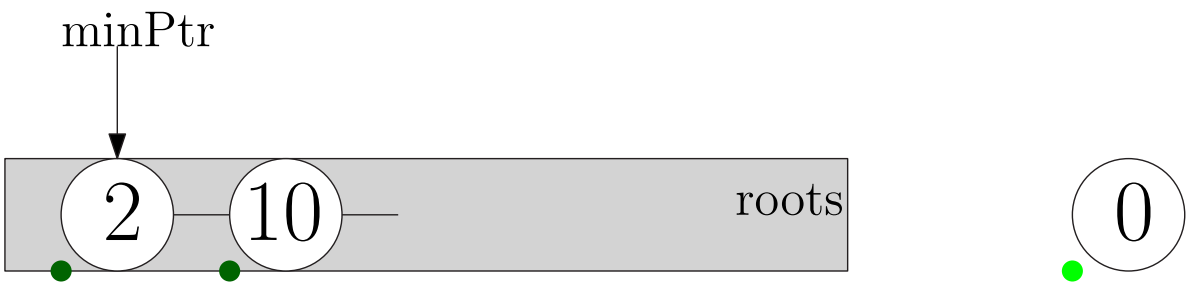
Fibonacci Heaps - Insert



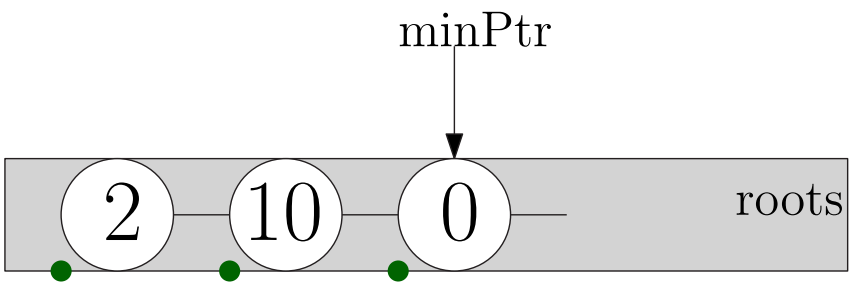
Fibonacci Heaps - Insert



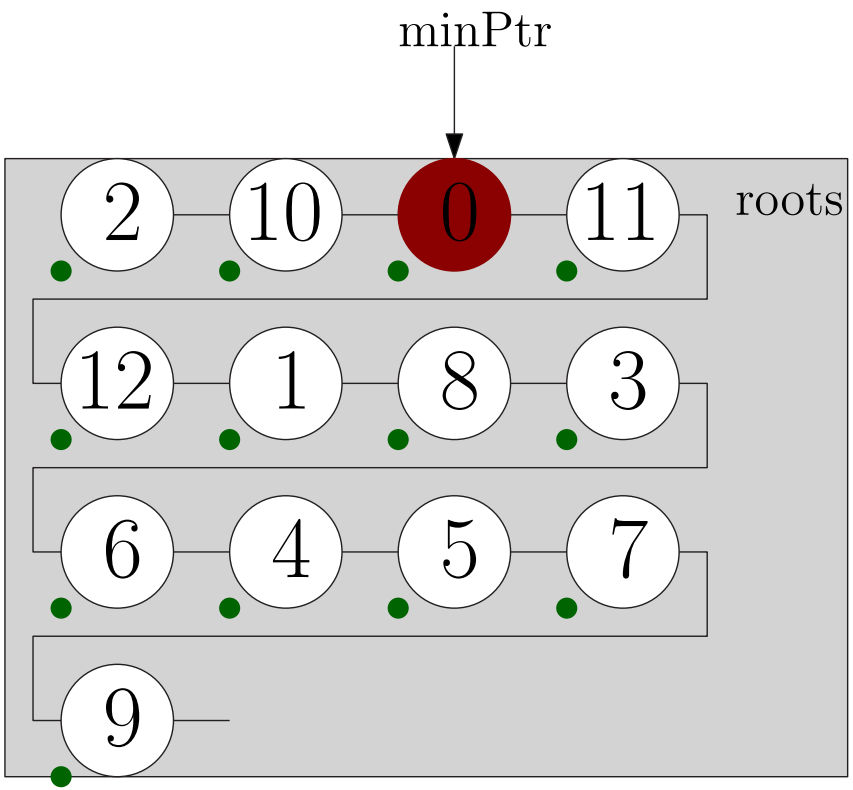
Fibonacci Heaps - Insert



Fibonacci Heaps - Insert



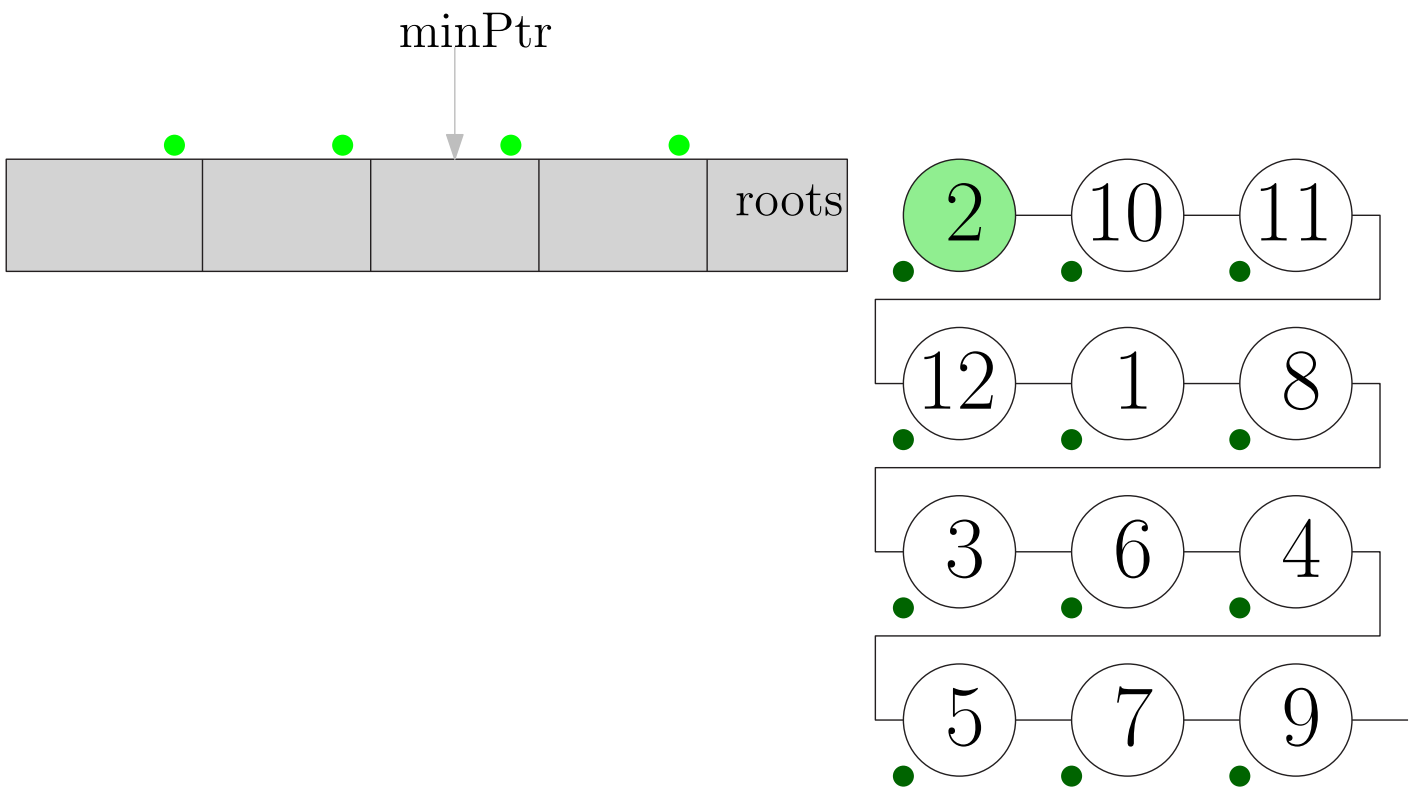
Fibonacci Heaps - Insert



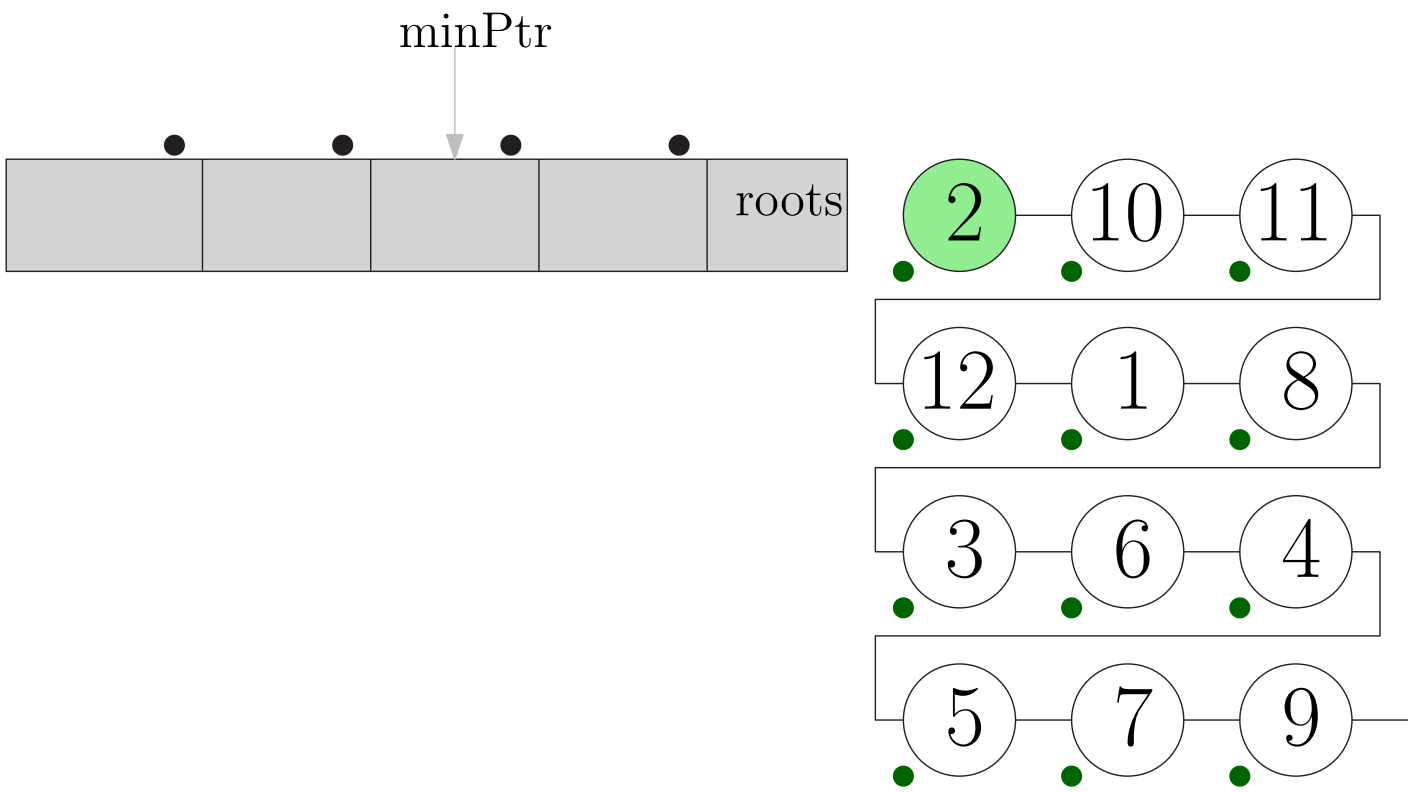
Fibonacci Heaps - Insert

- Insert in konstanter Zeit
- Erzeugt großen Wald einzelner Knoten
- Entspricht ohne **union** nur linearer Liste

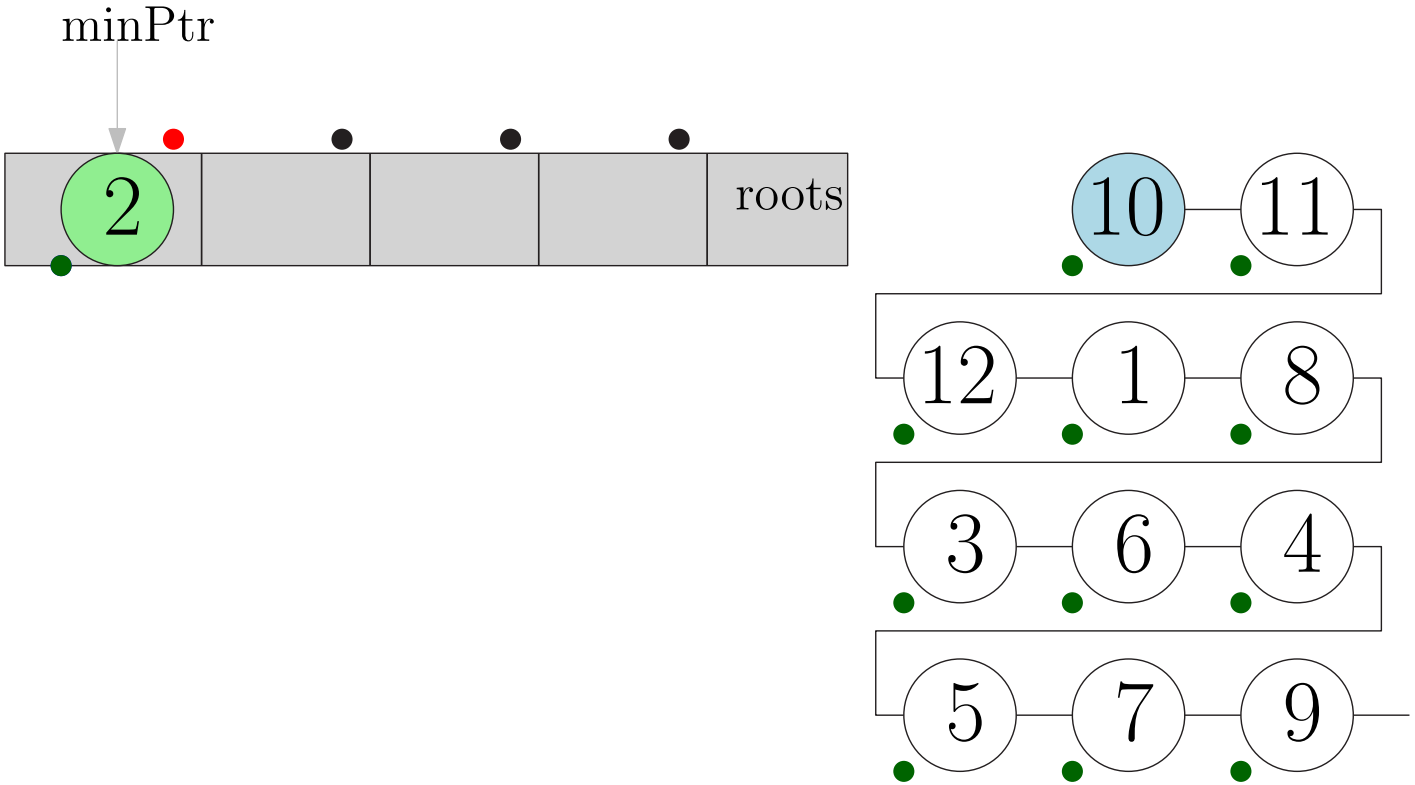
Fibonacci Heaps - Delete Min



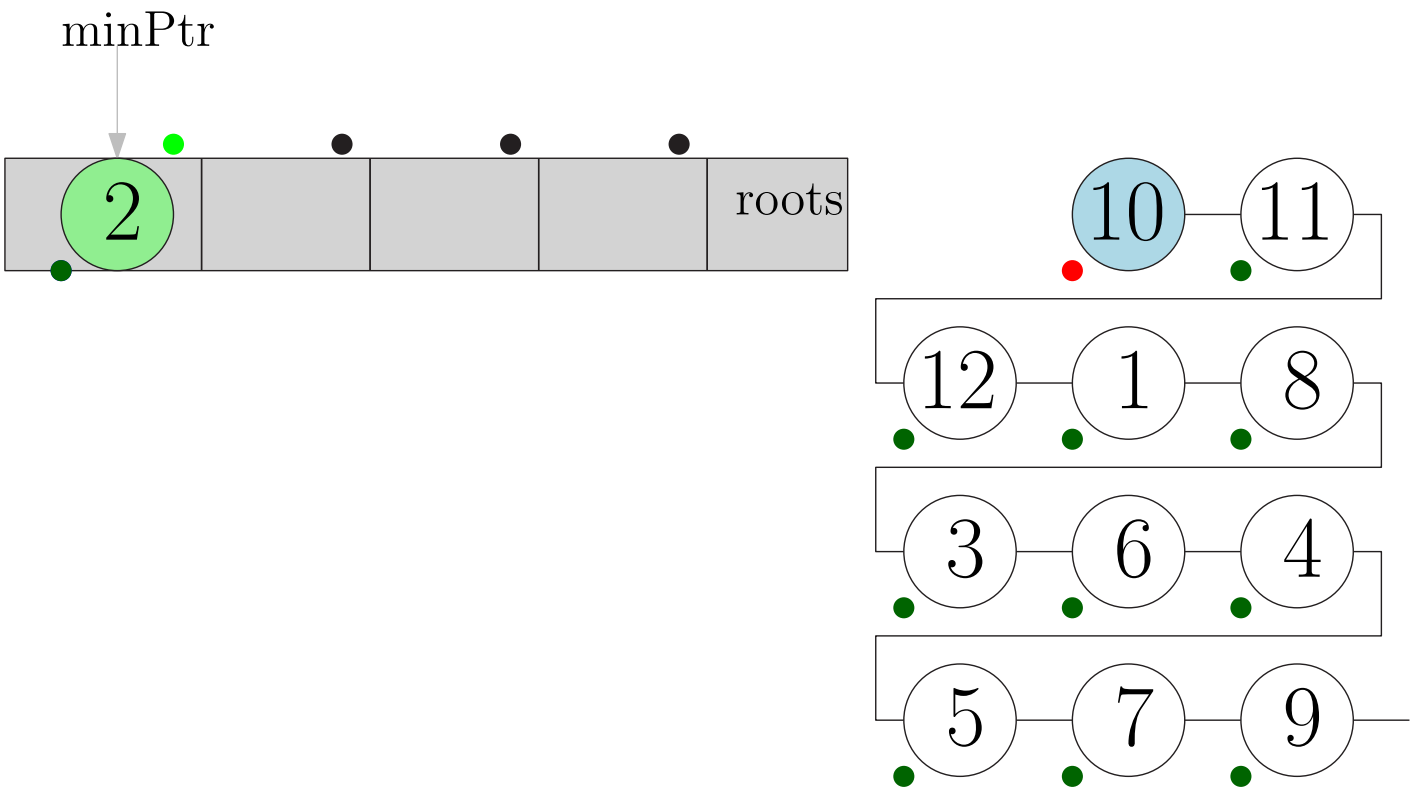
Fibonacci Heaps - Delete Min



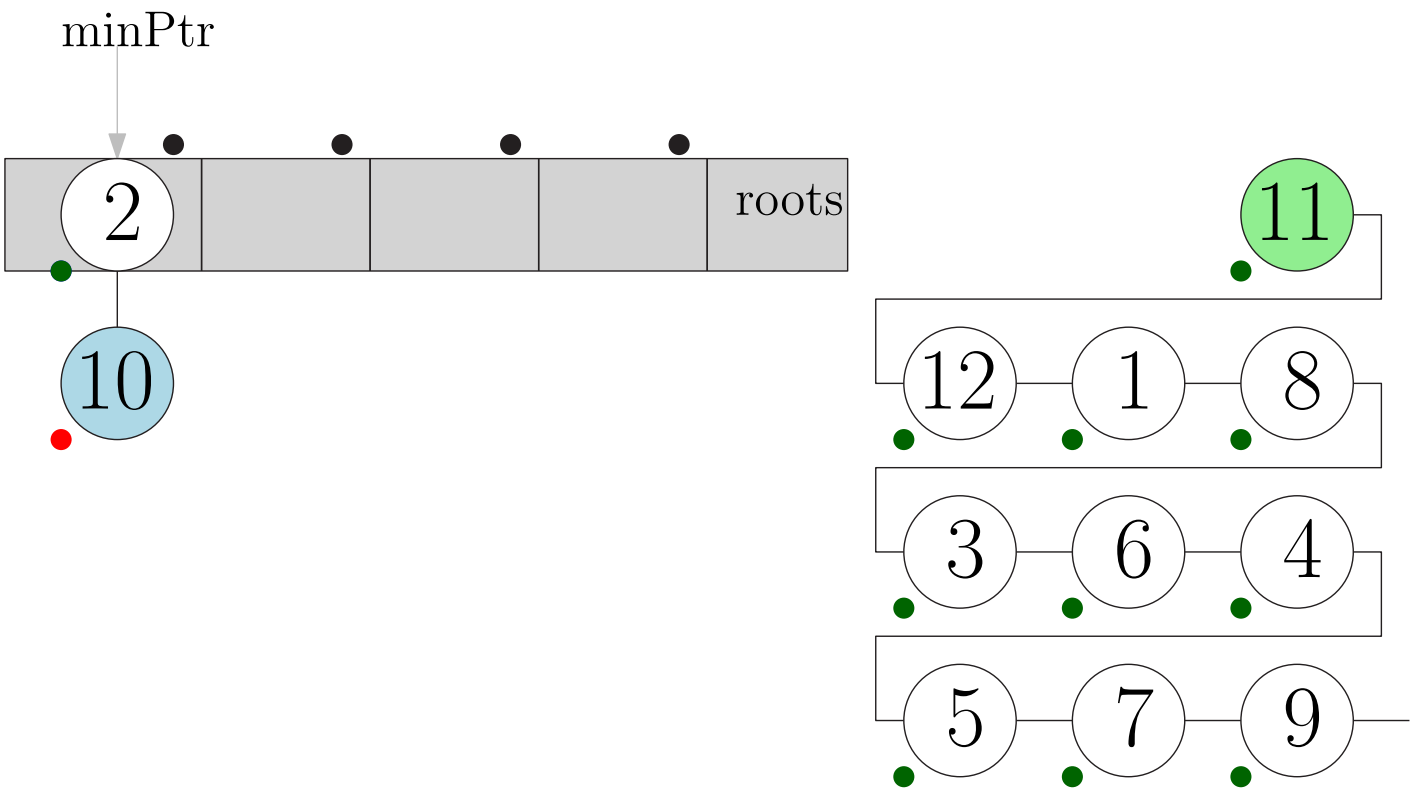
Fibonacci Heaps - Delete Min



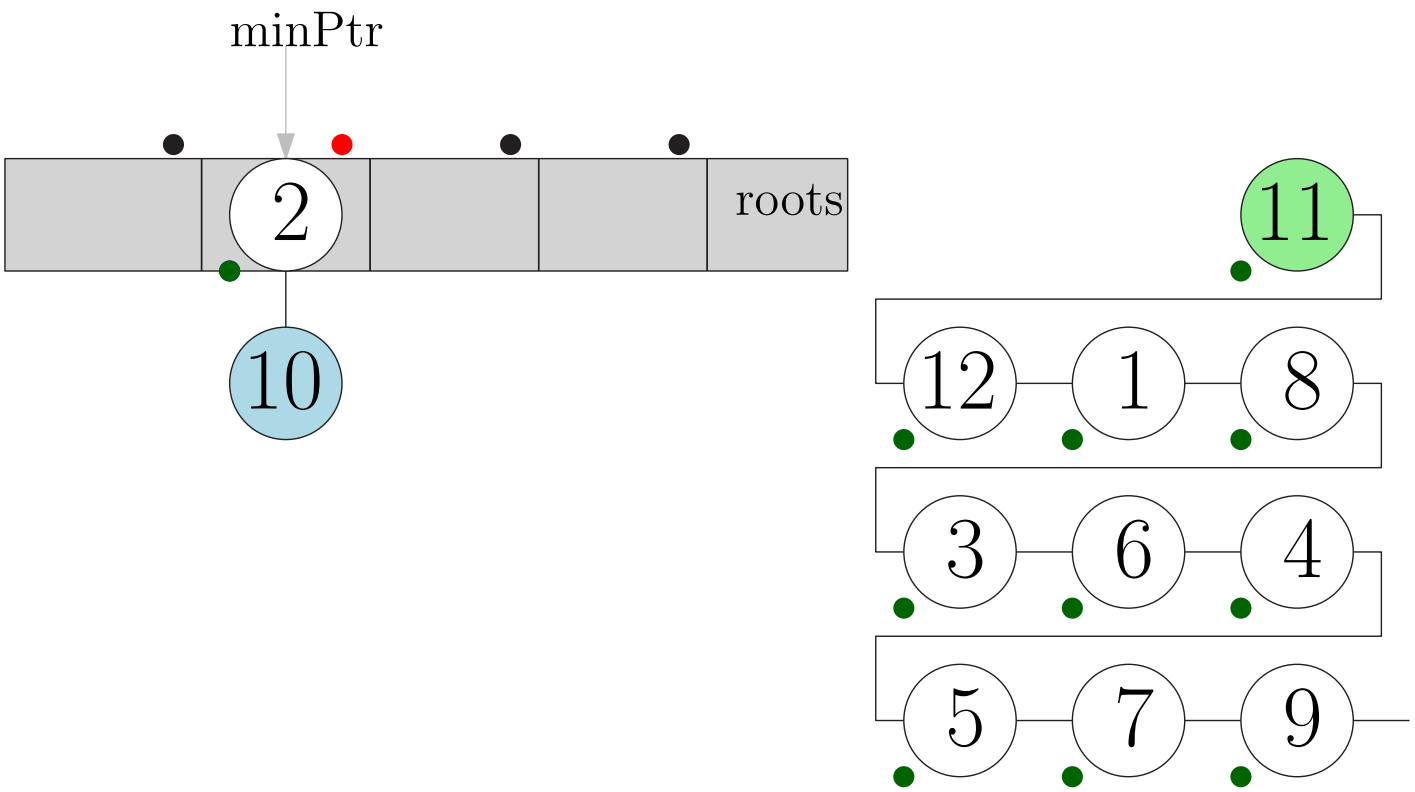
Fibonacci Heaps - Delete Min



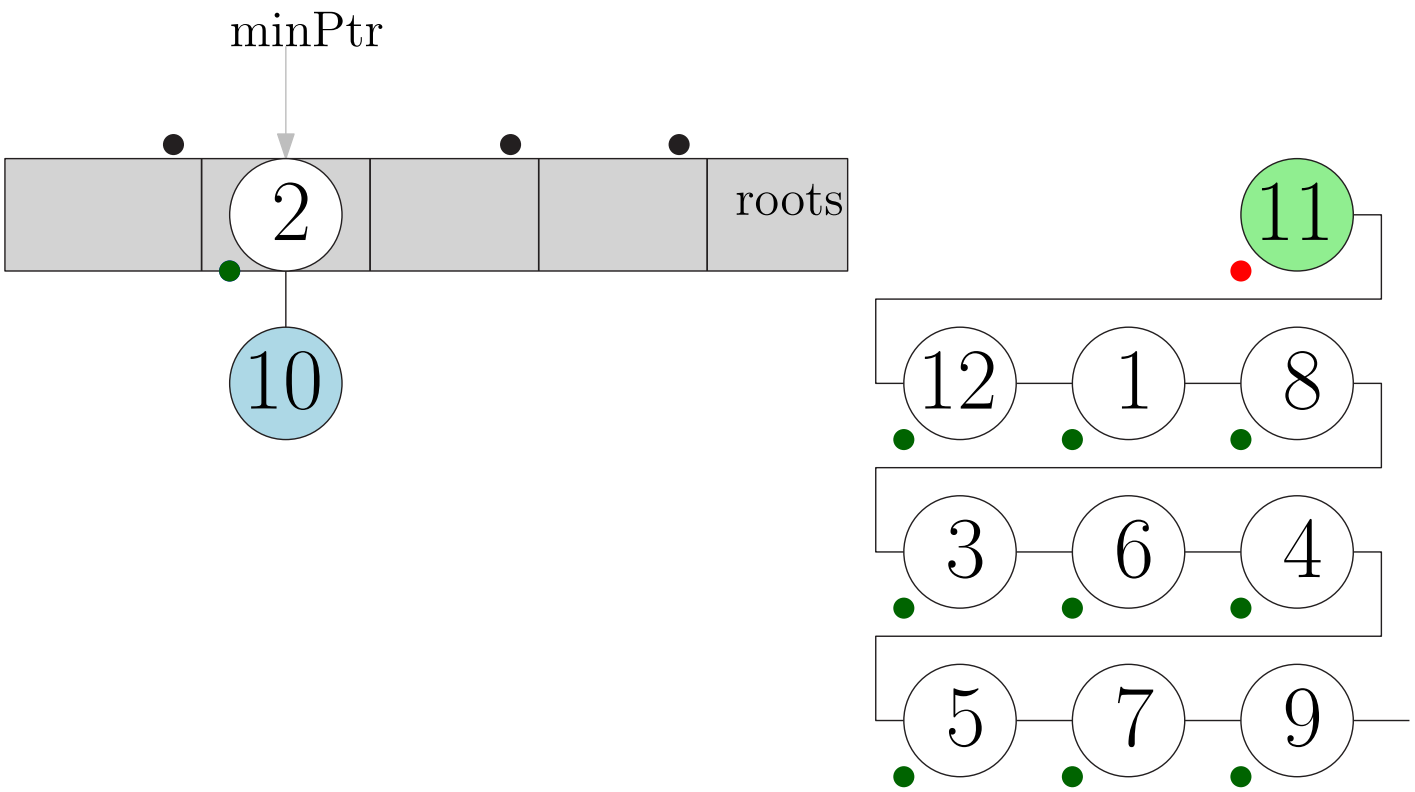
Fibonacci Heaps - Delete Min



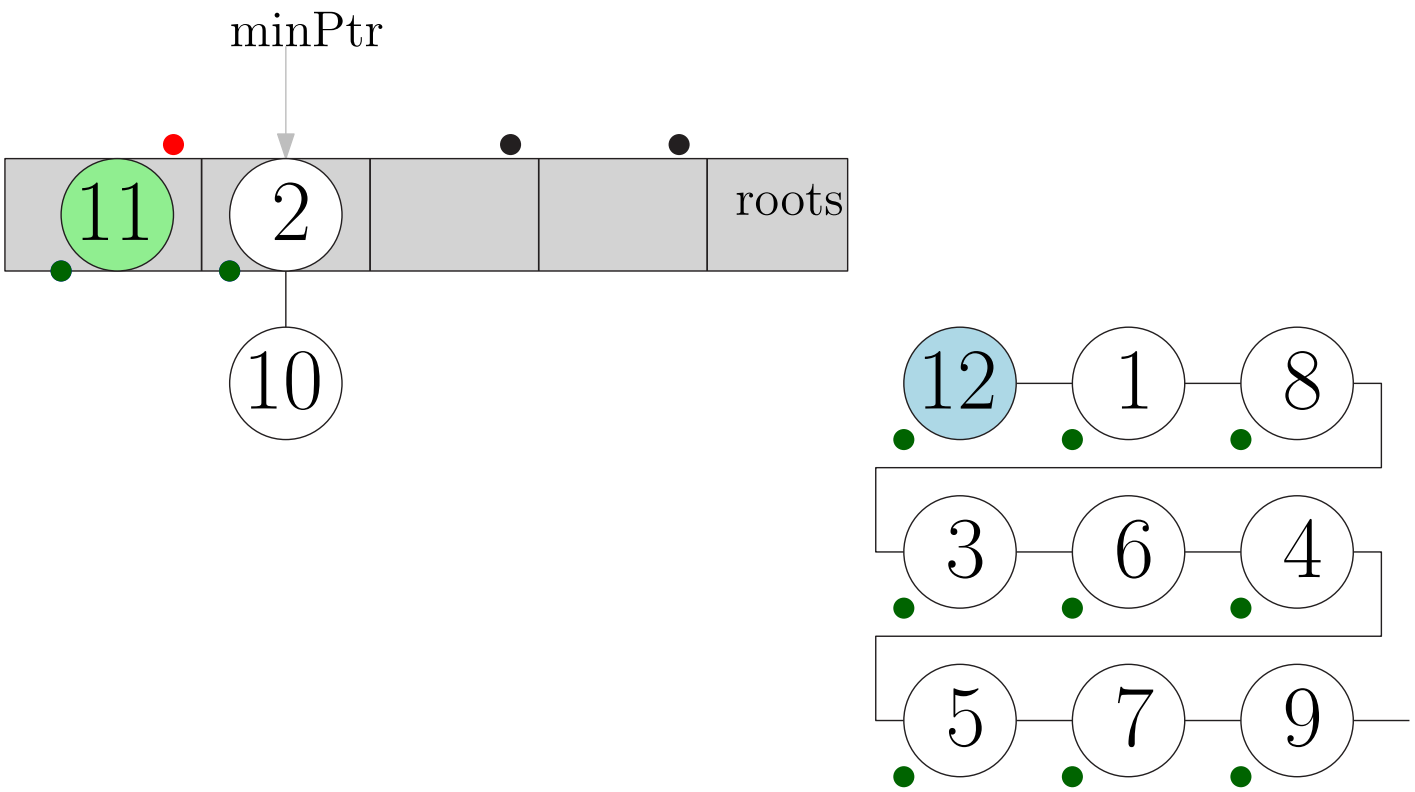
Fibonacci Heaps - Delete Min



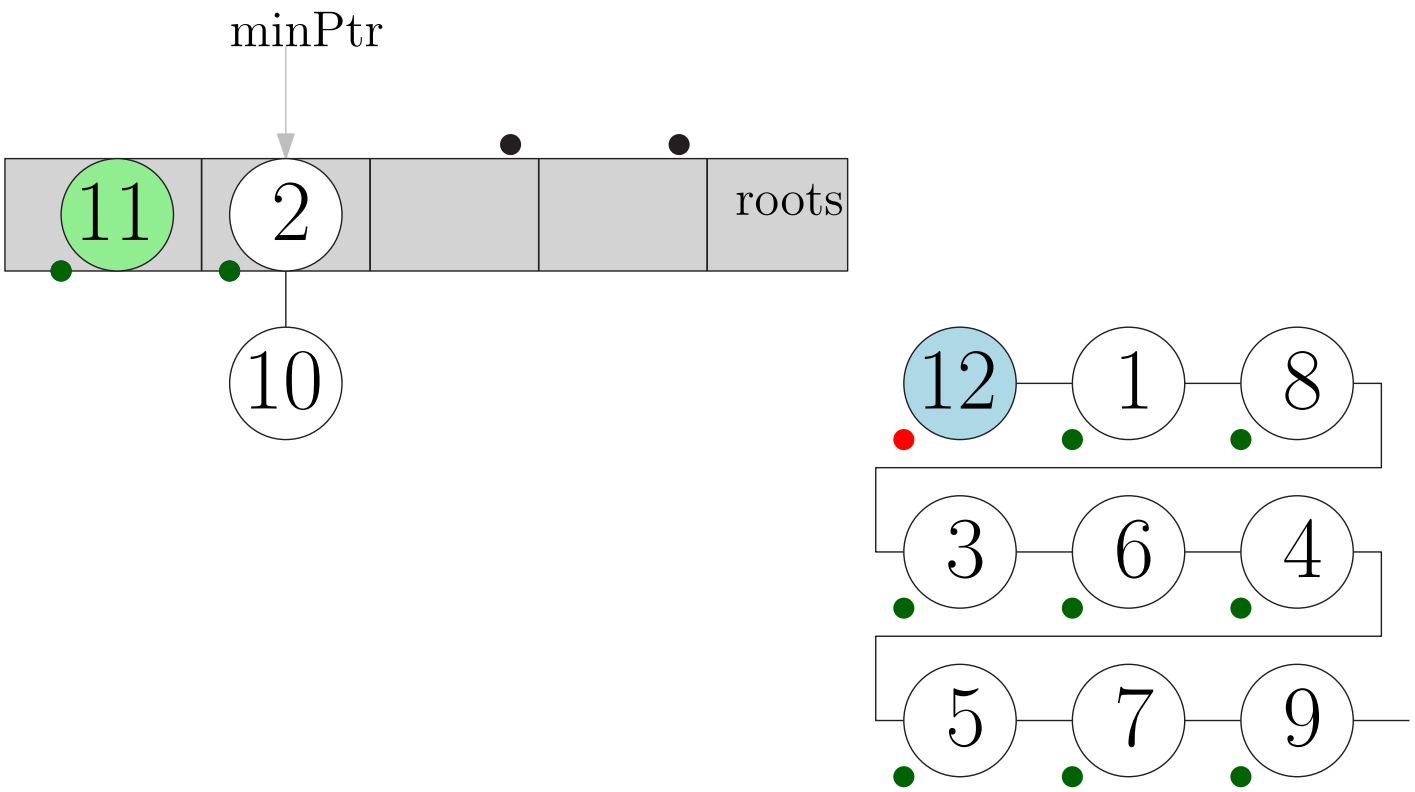
Fibonacci Heaps - Delete Min



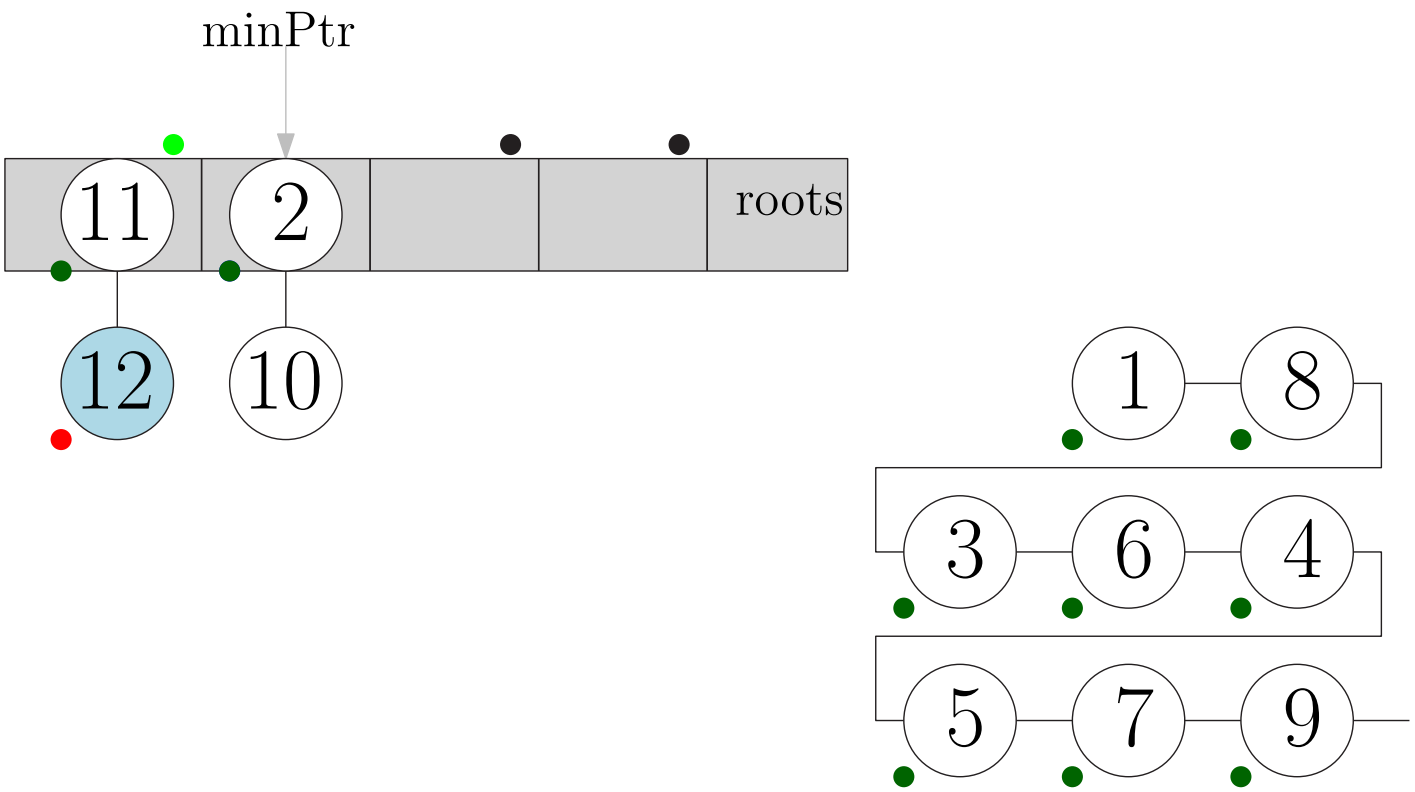
Fibonacci Heaps - Delete Min



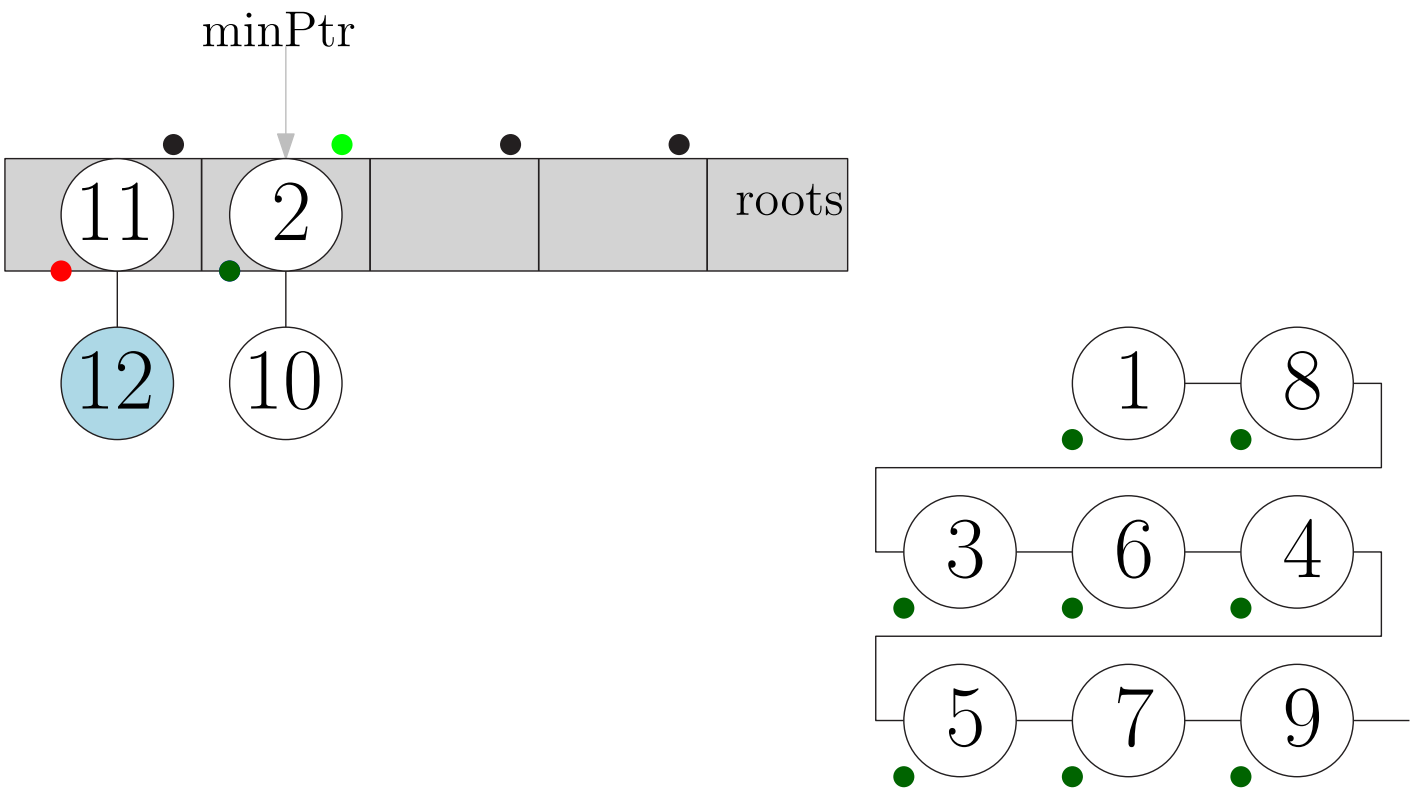
Fibonacci Heaps - Delete Min



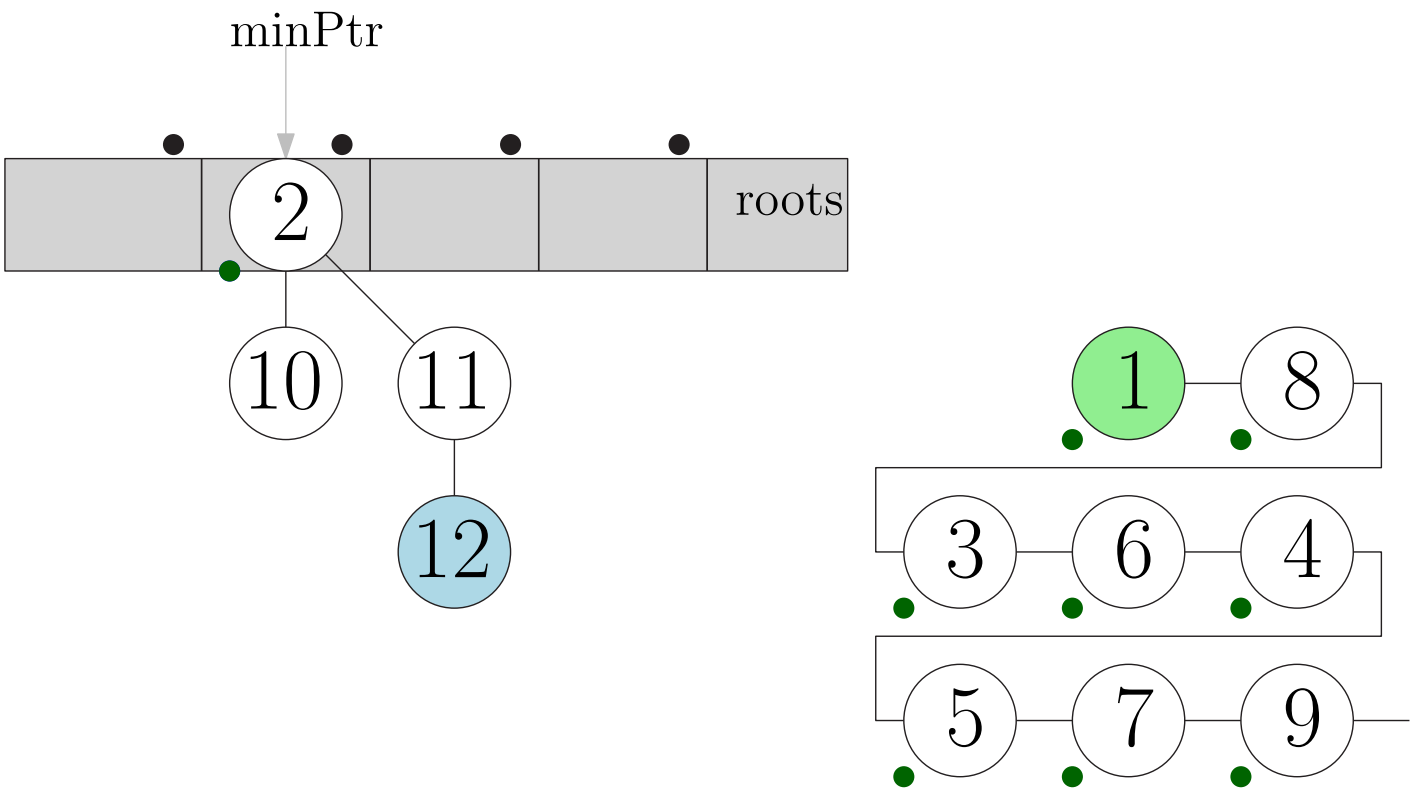
Fibonacci Heaps - Delete Min



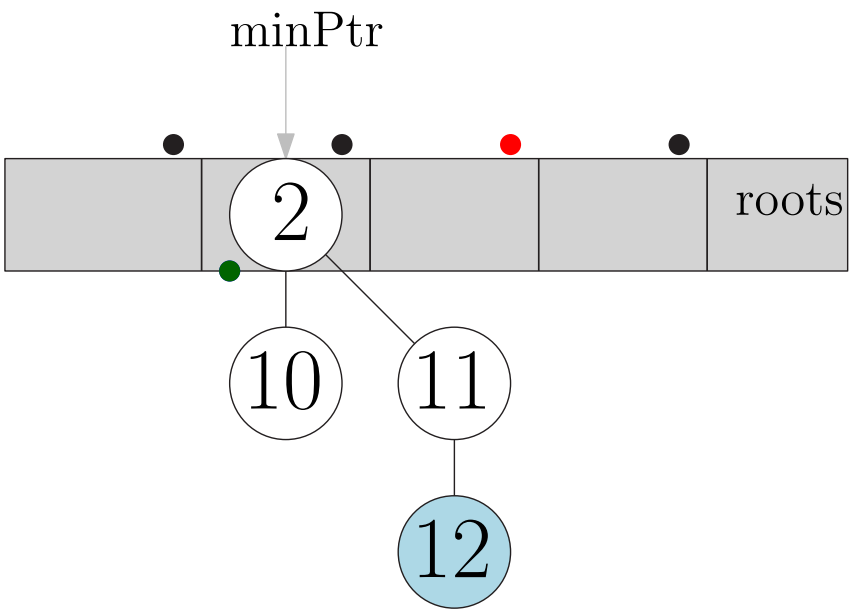
Fibonacci Heaps - Delete Min

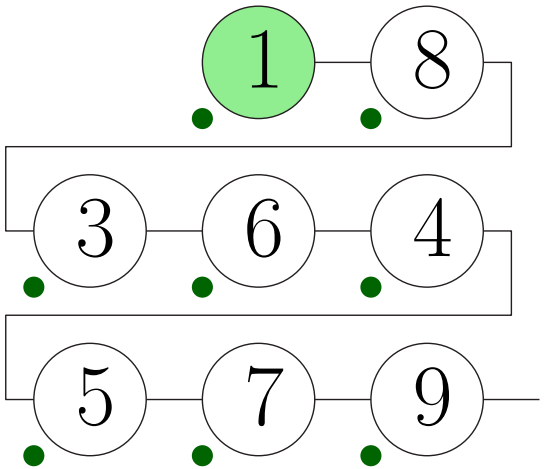


Fibonacci Heaps - Delete Min

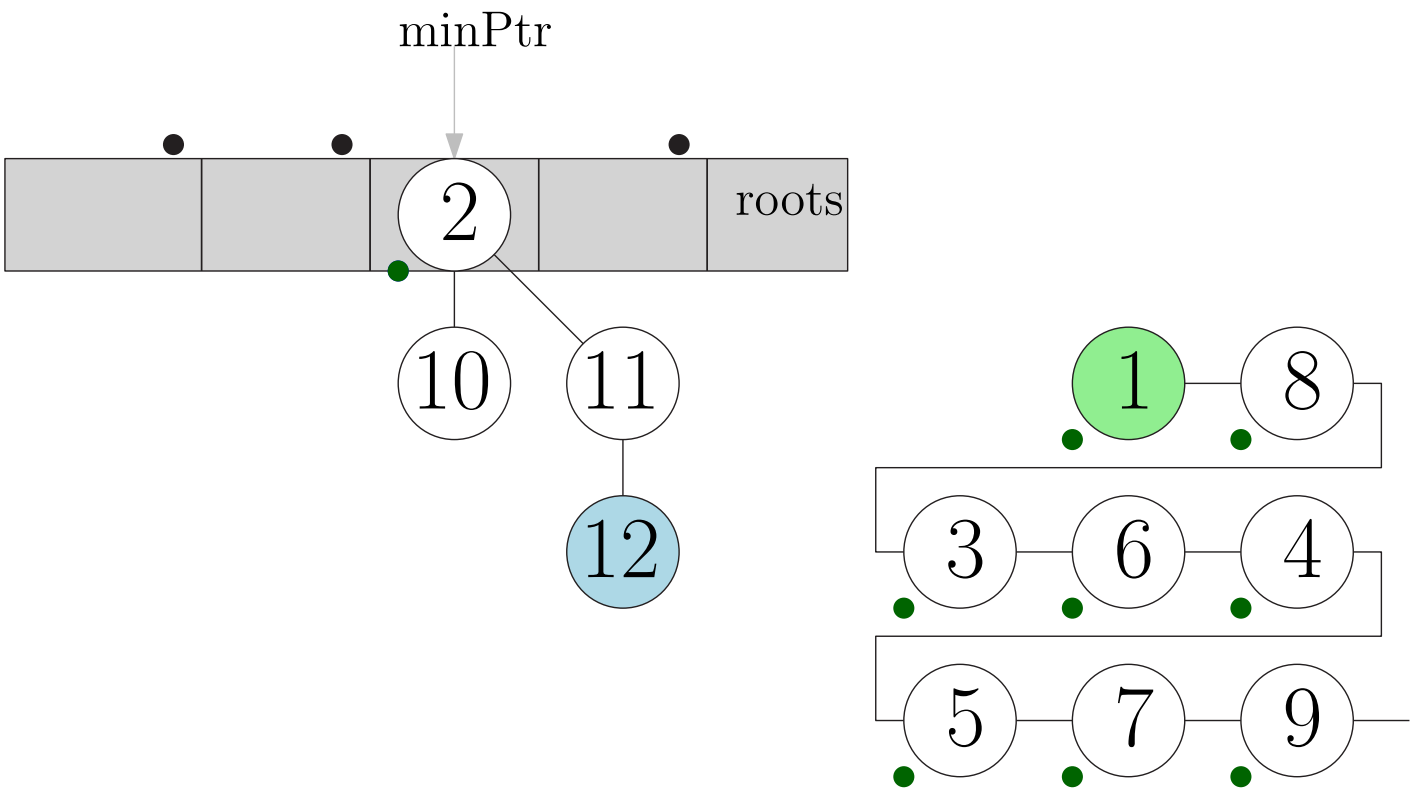


Fibonacci Heaps - Delete Min

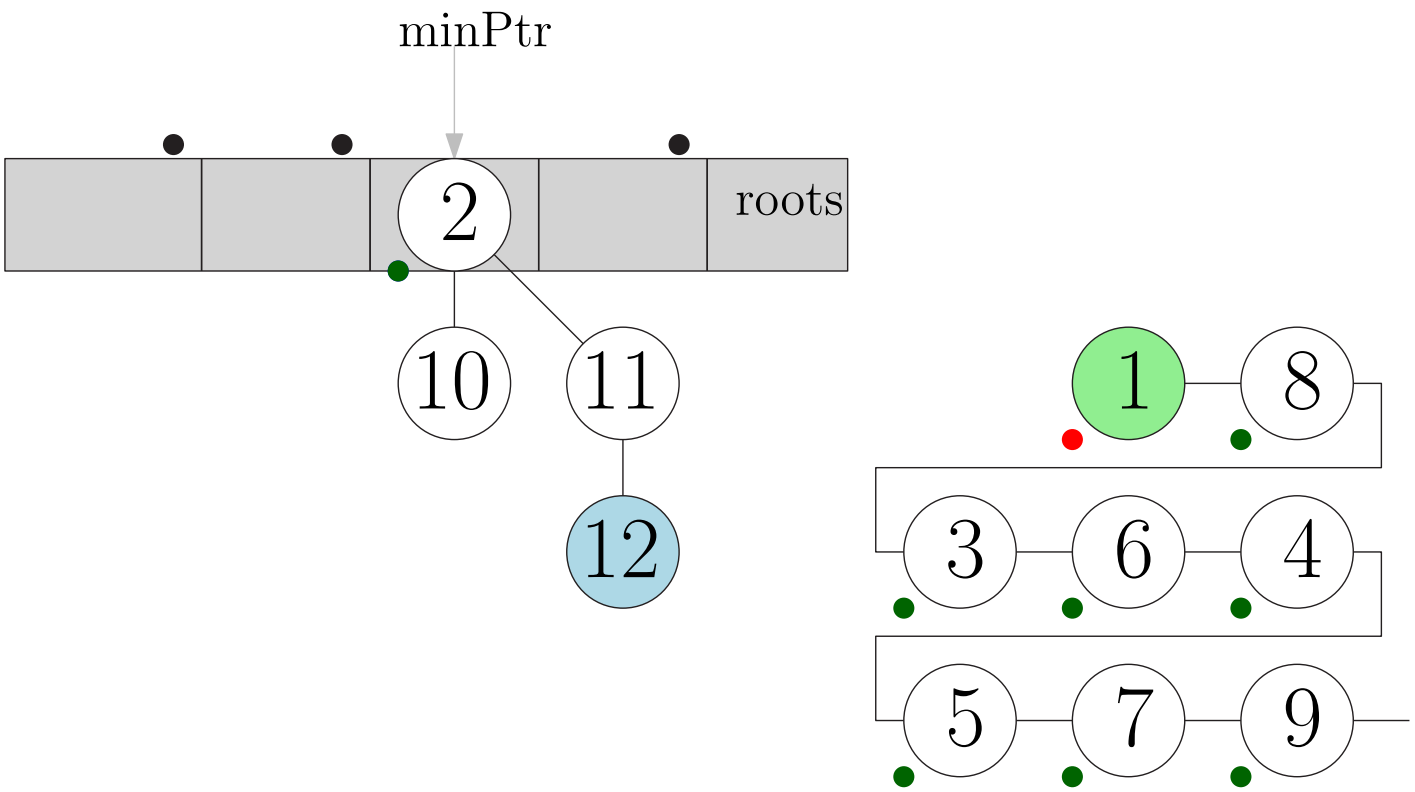




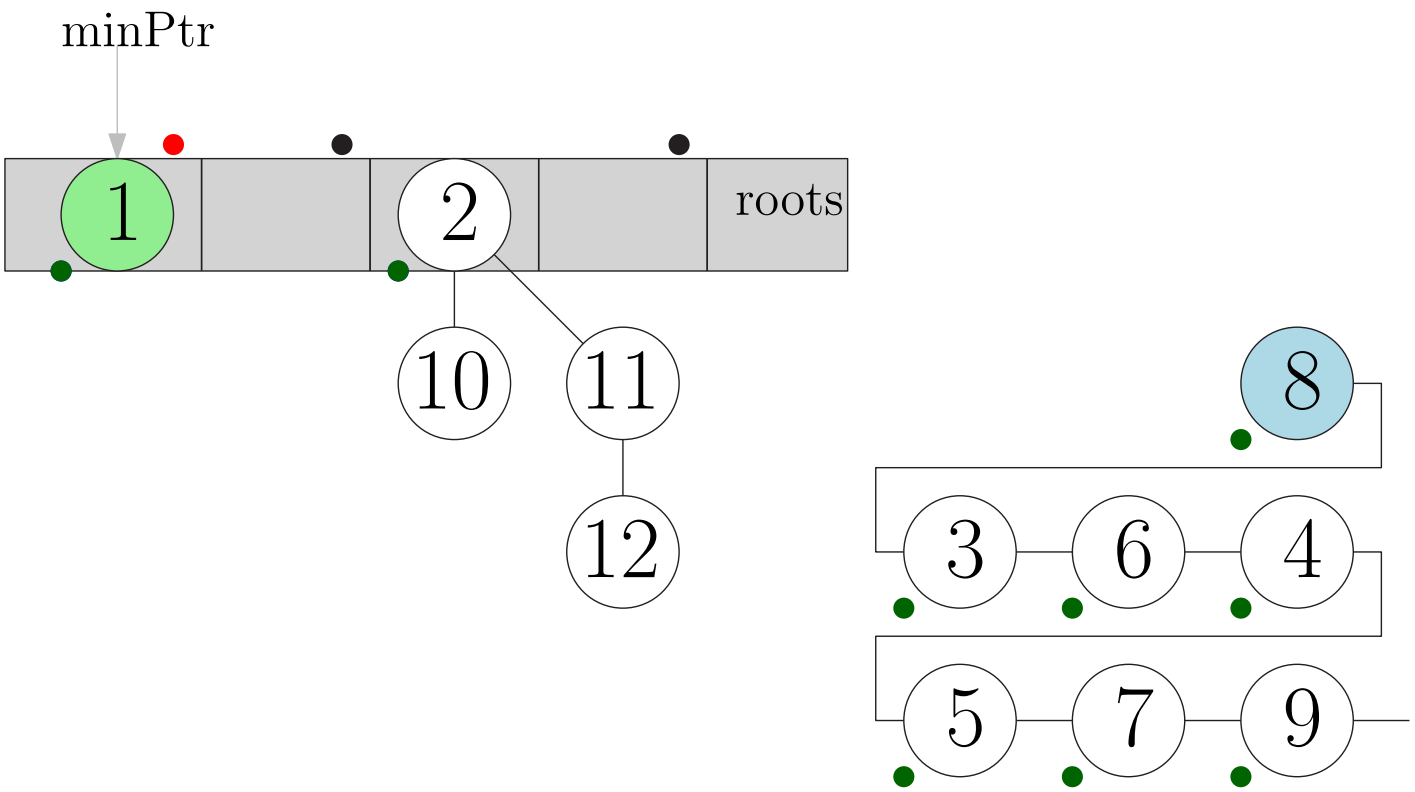
Fibonacci Heaps - Delete Min



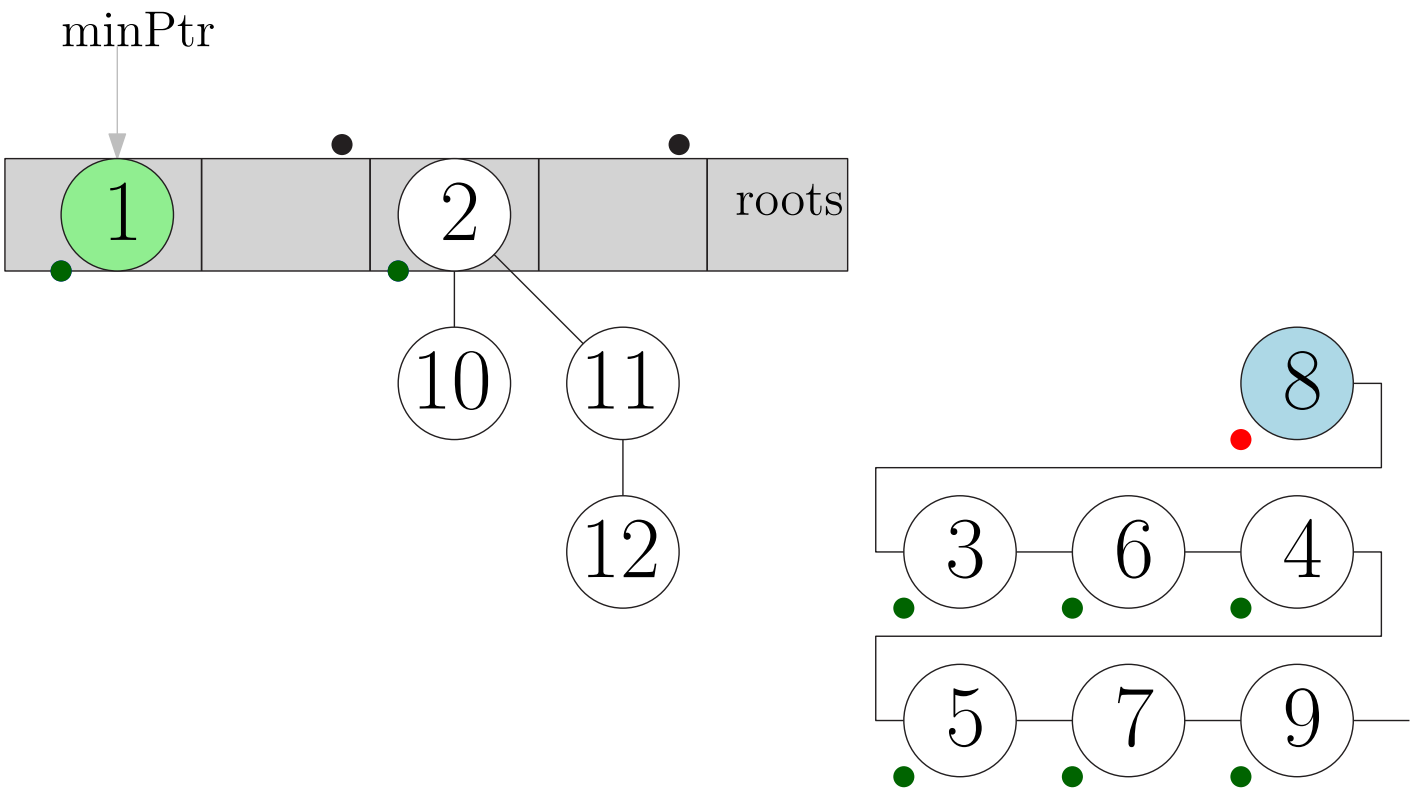
Fibonacci Heaps - Delete Min



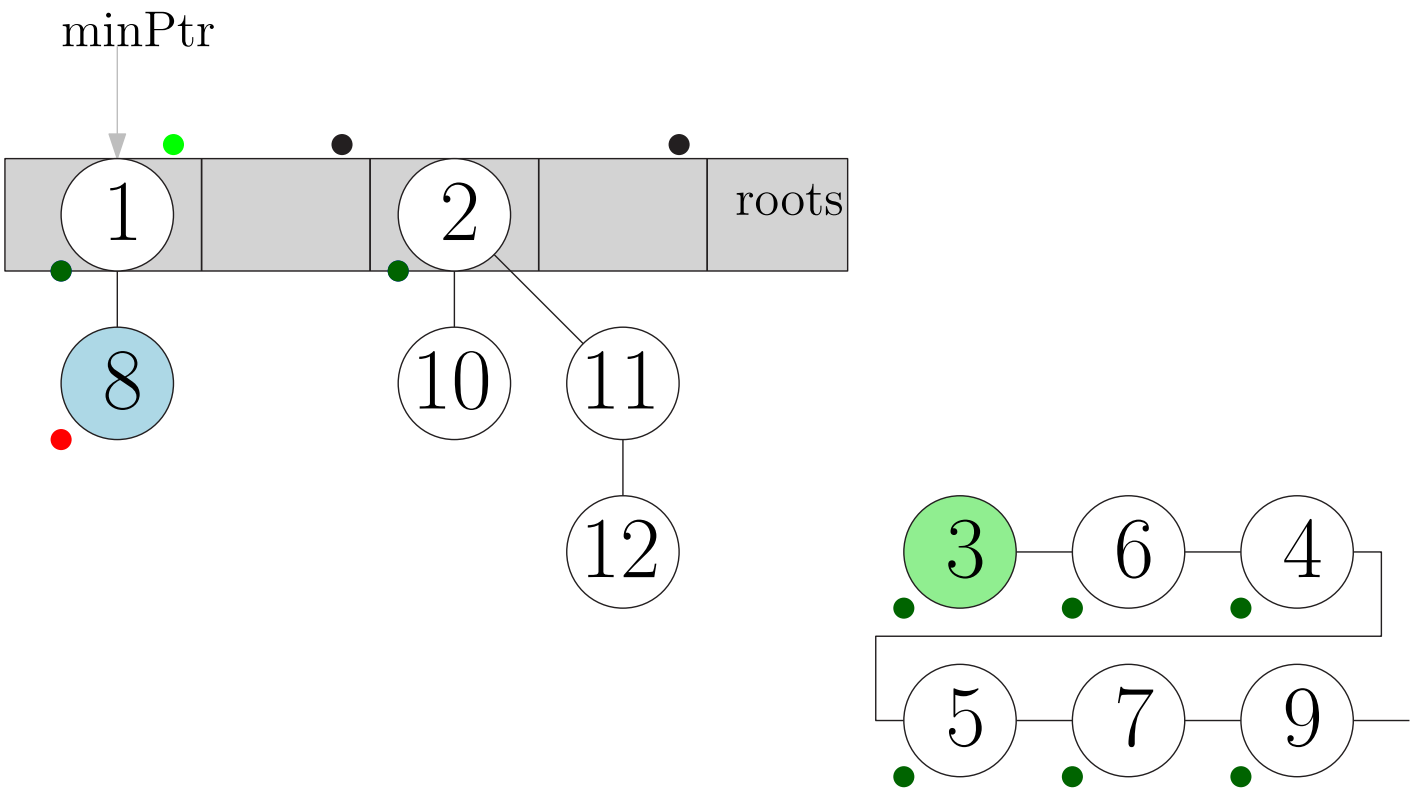
Fibonacci Heaps - Delete Min



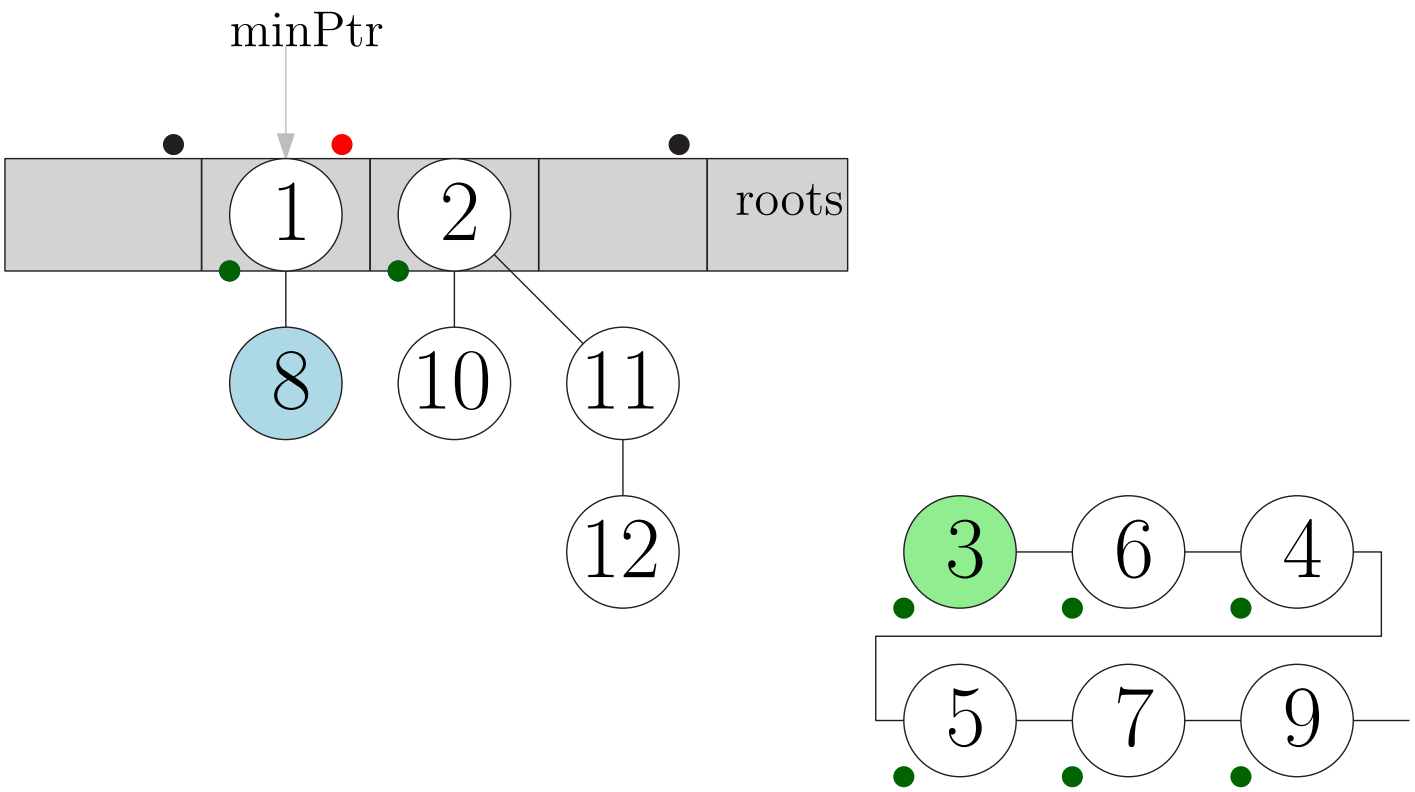
Fibonacci Heaps - Delete Min



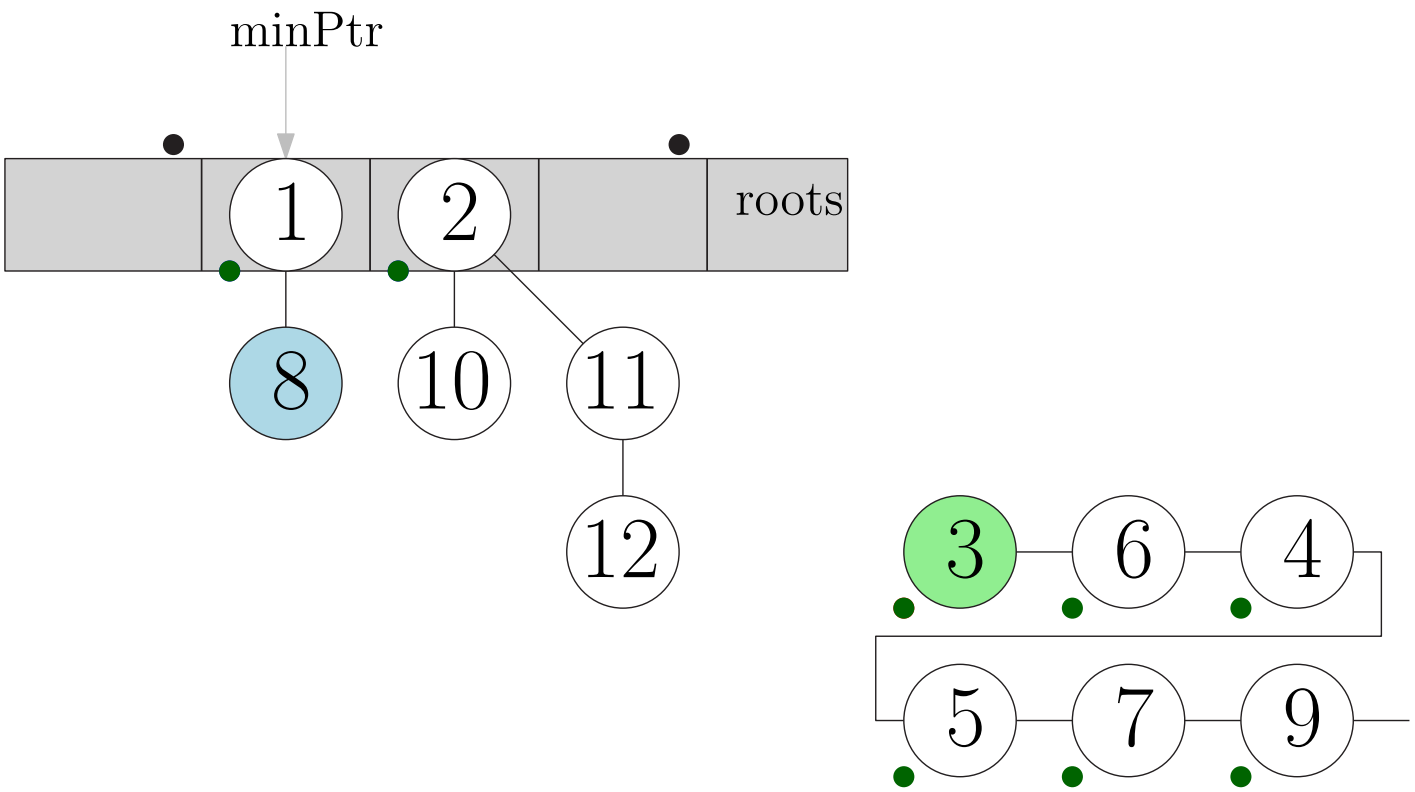
Fibonacci Heaps - Delete Min



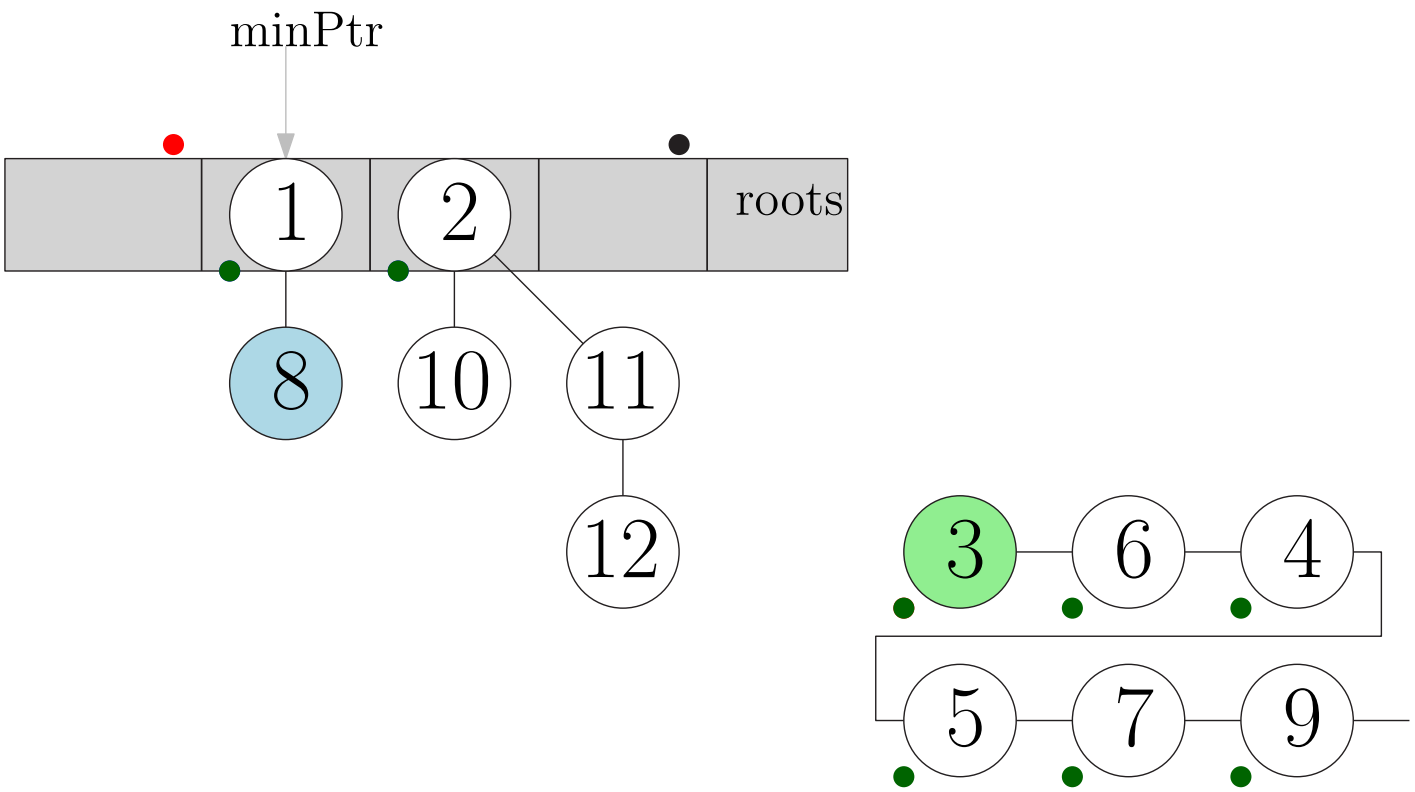
Fibonacci Heaps - Delete Min



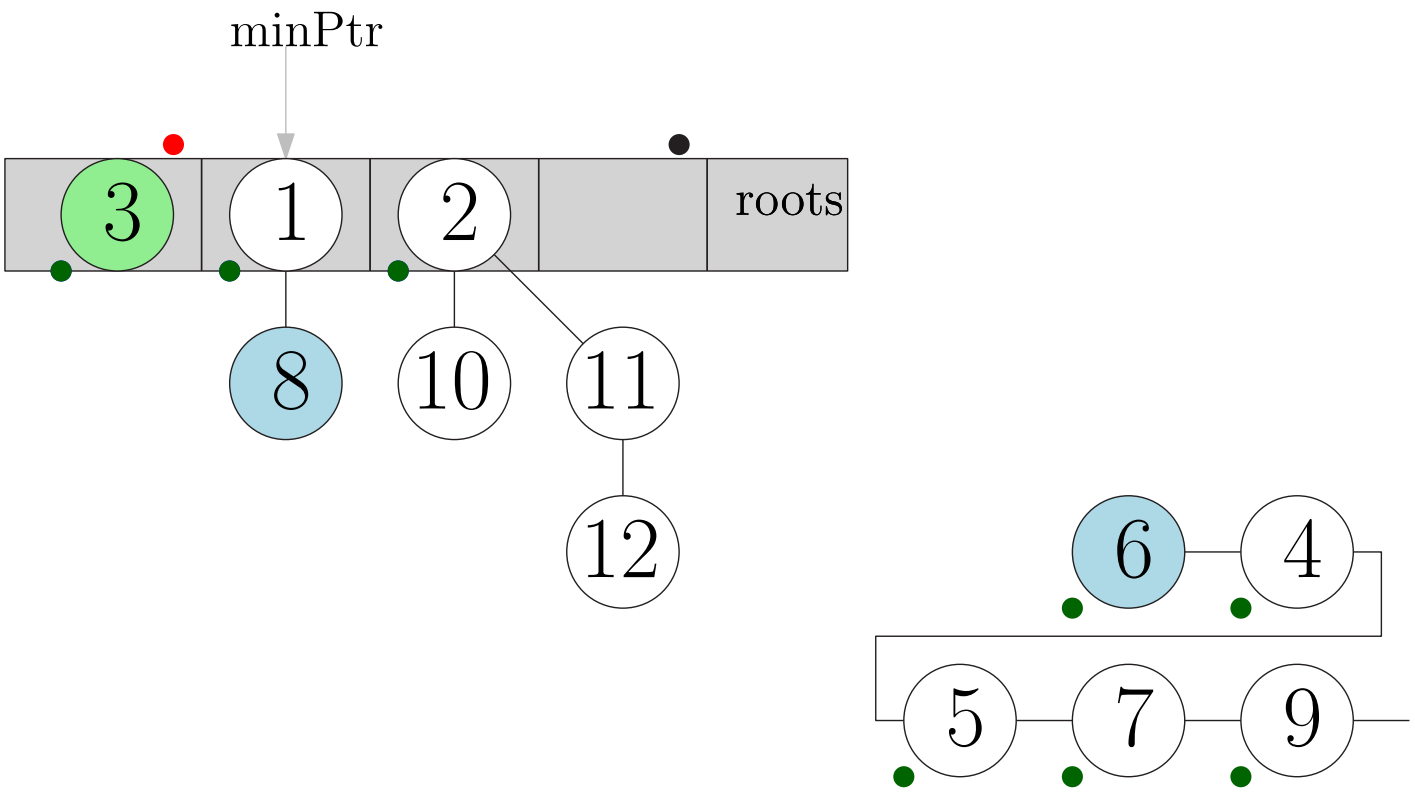
Fibonacci Heaps - Delete Min



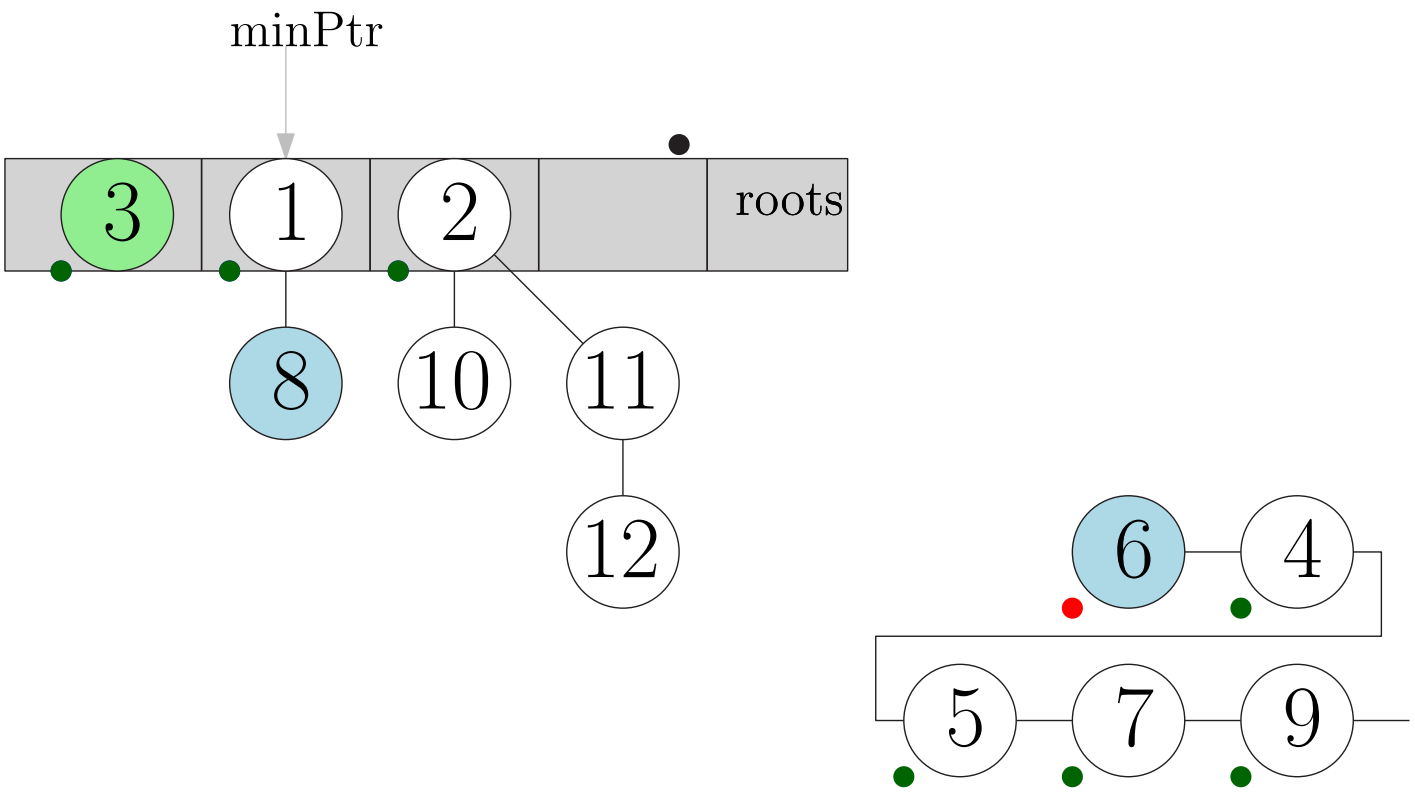
Fibonacci Heaps - Delete Min



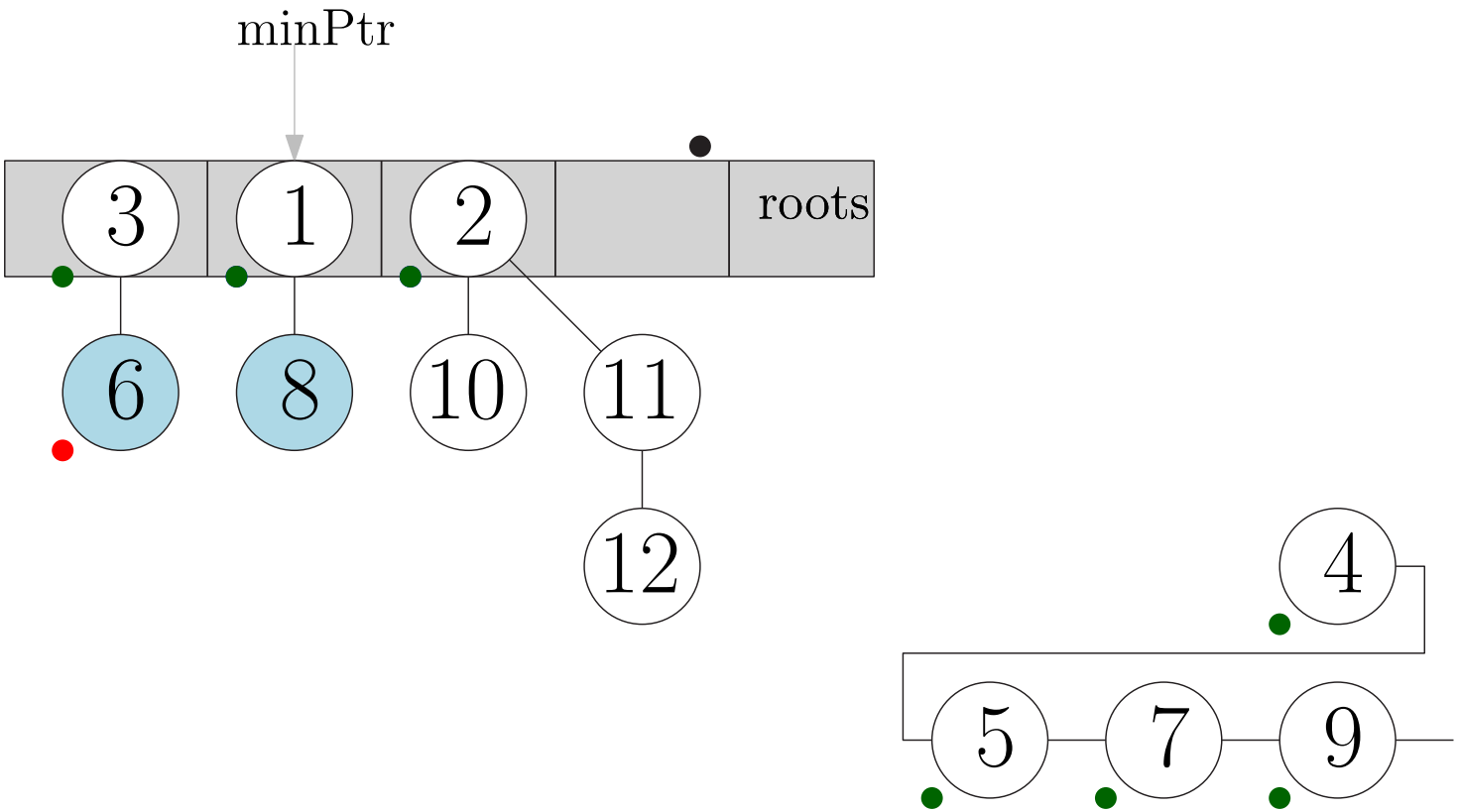
Fibonacci Heaps - Delete Min



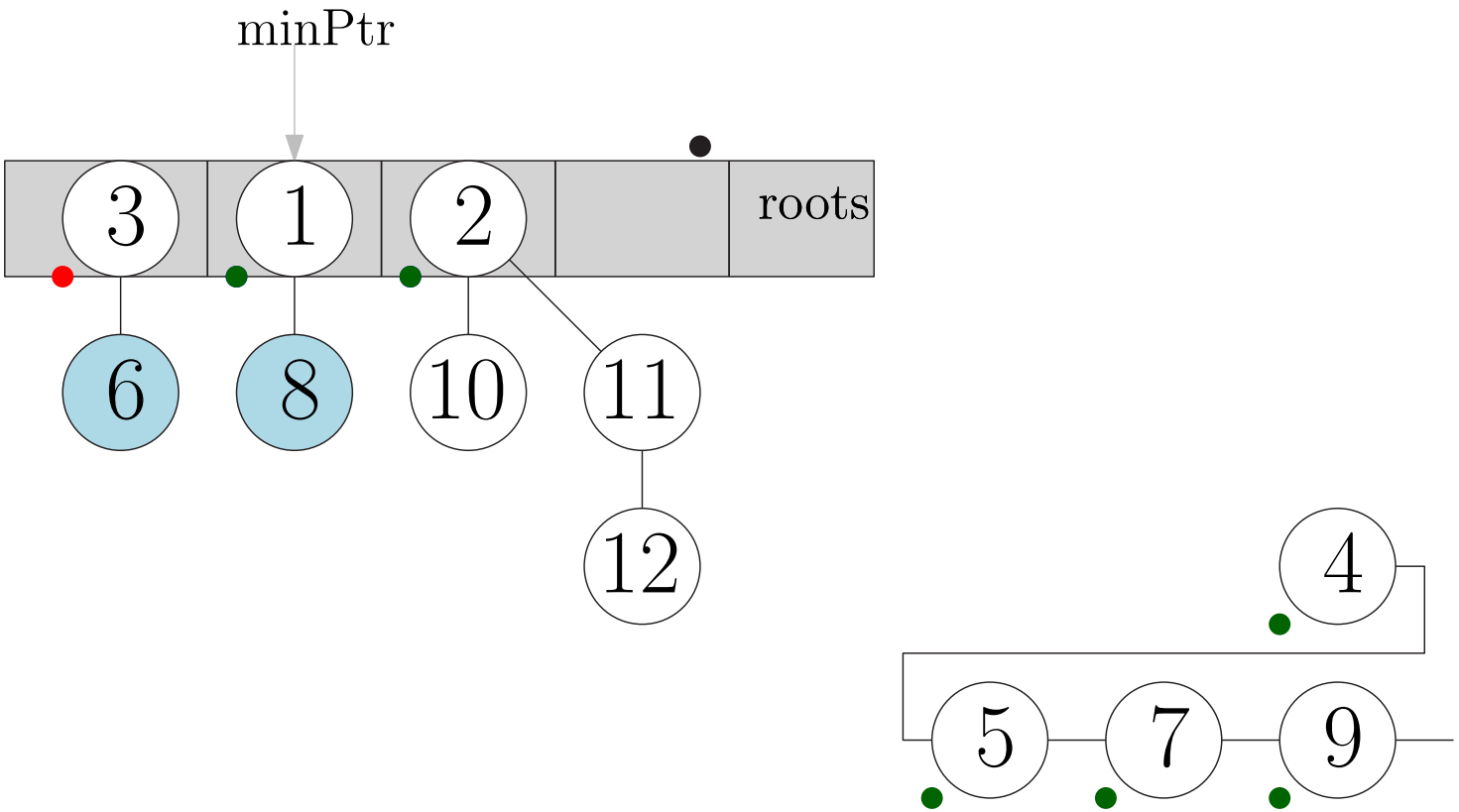
Fibonacci Heaps - Delete Min



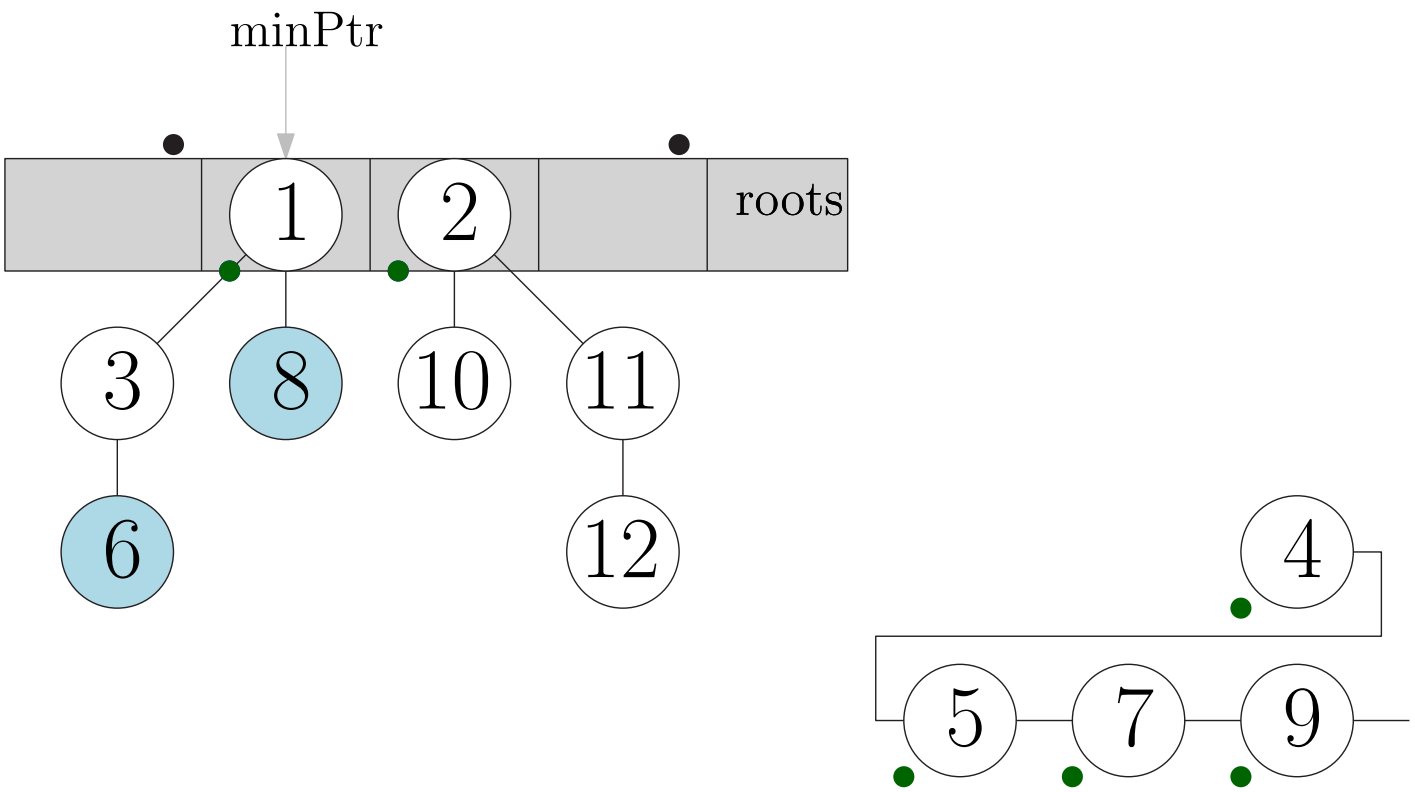
Fibonacci Heaps - Delete Min



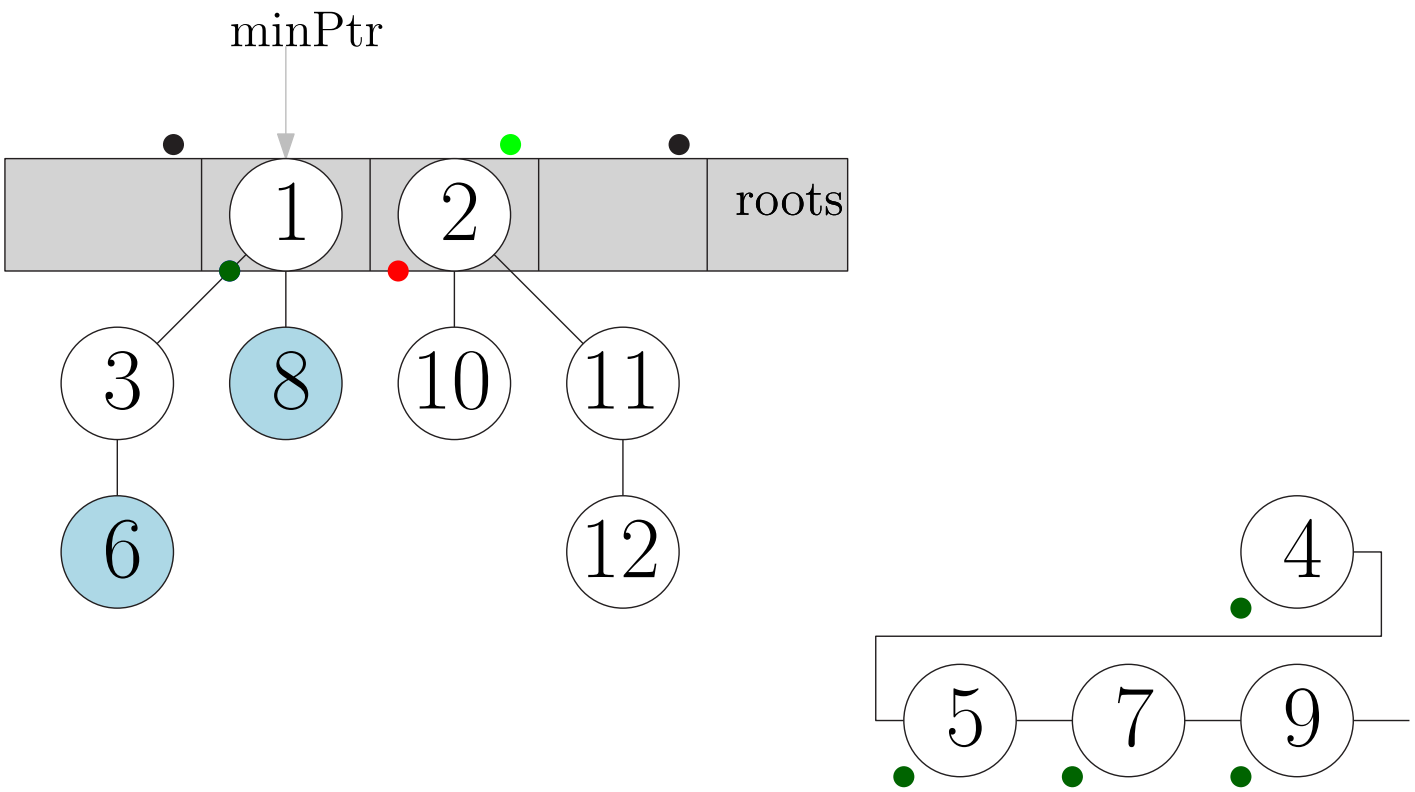
Fibonacci Heaps - Delete Min



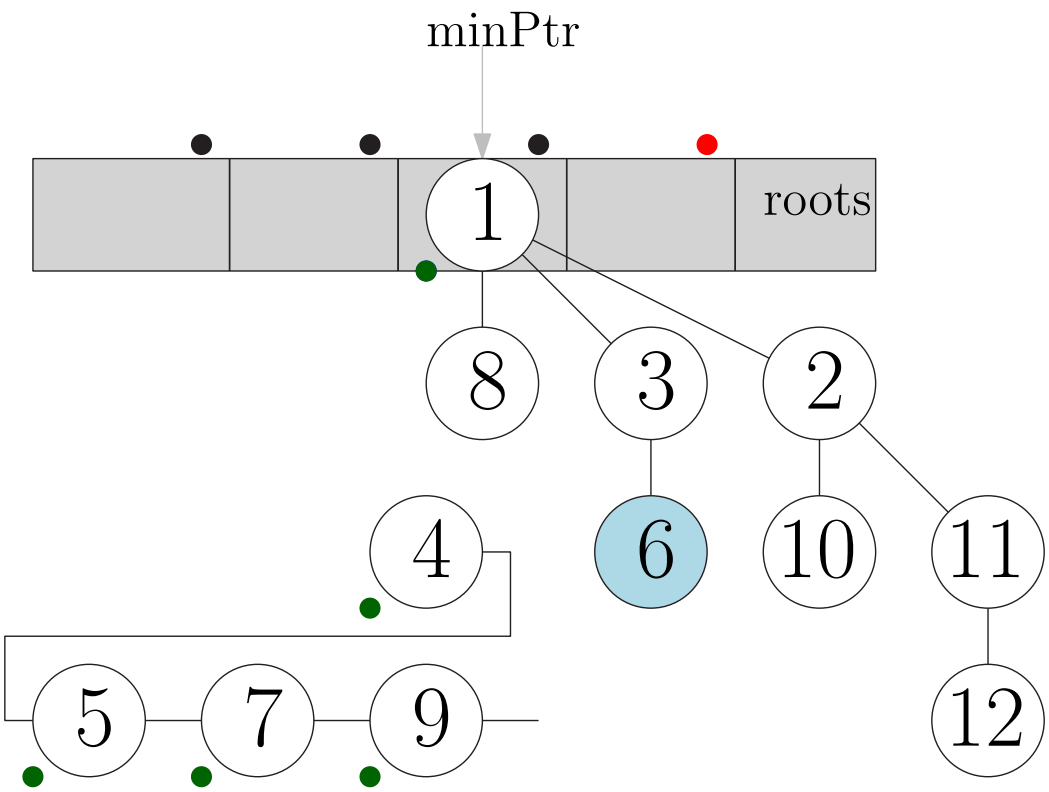
Fibonacci Heaps - Delete Min



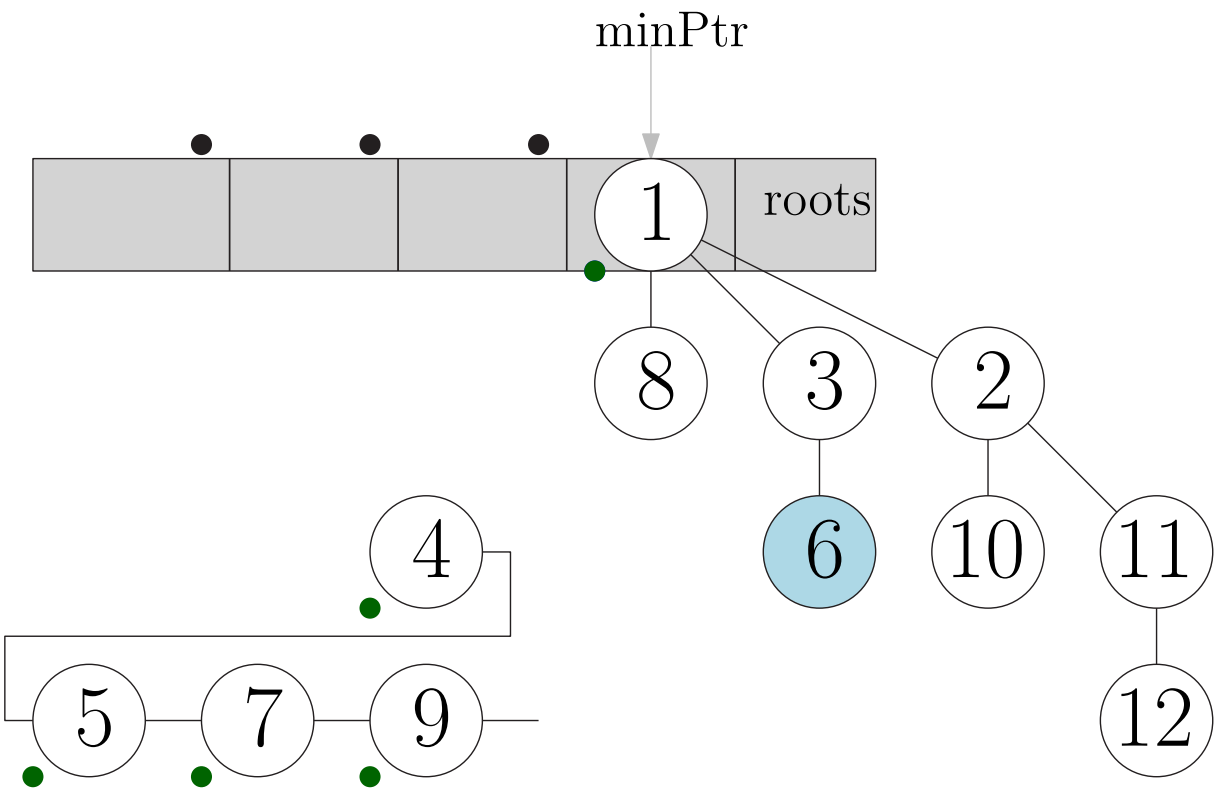
Fibonacci Heaps - Delete Min



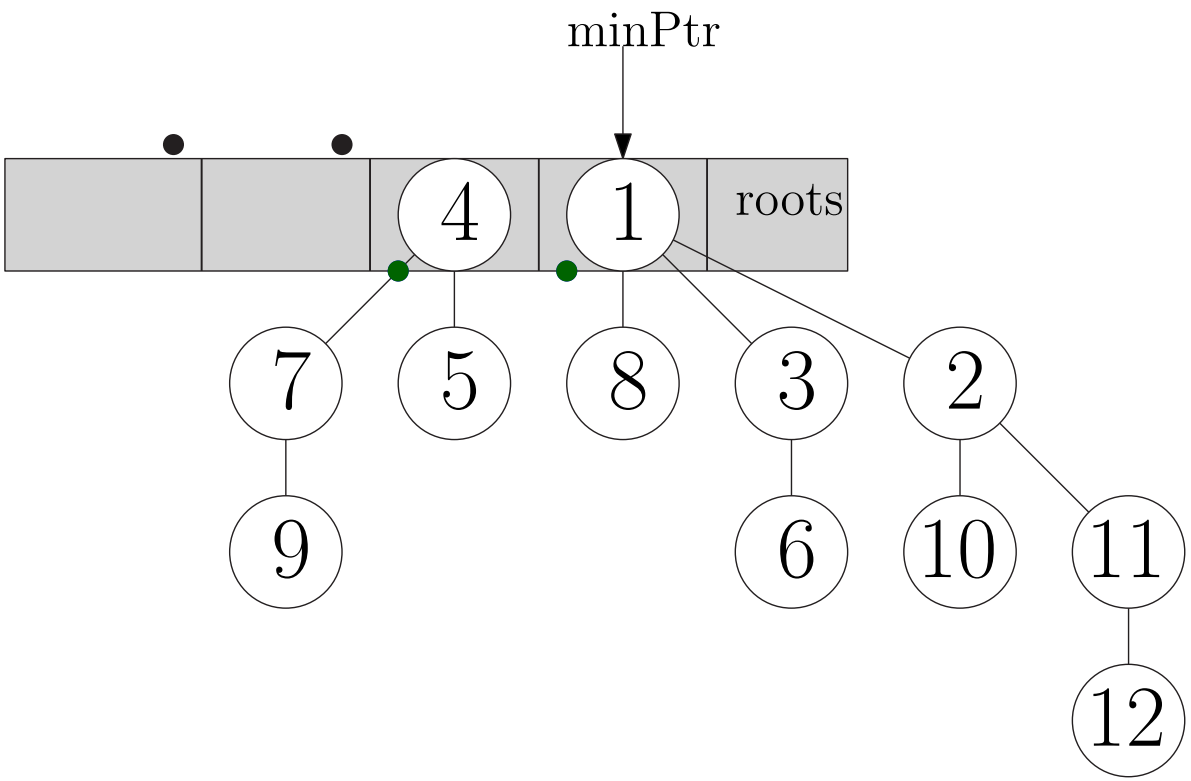
Fibonacci Heaps - Delete Min



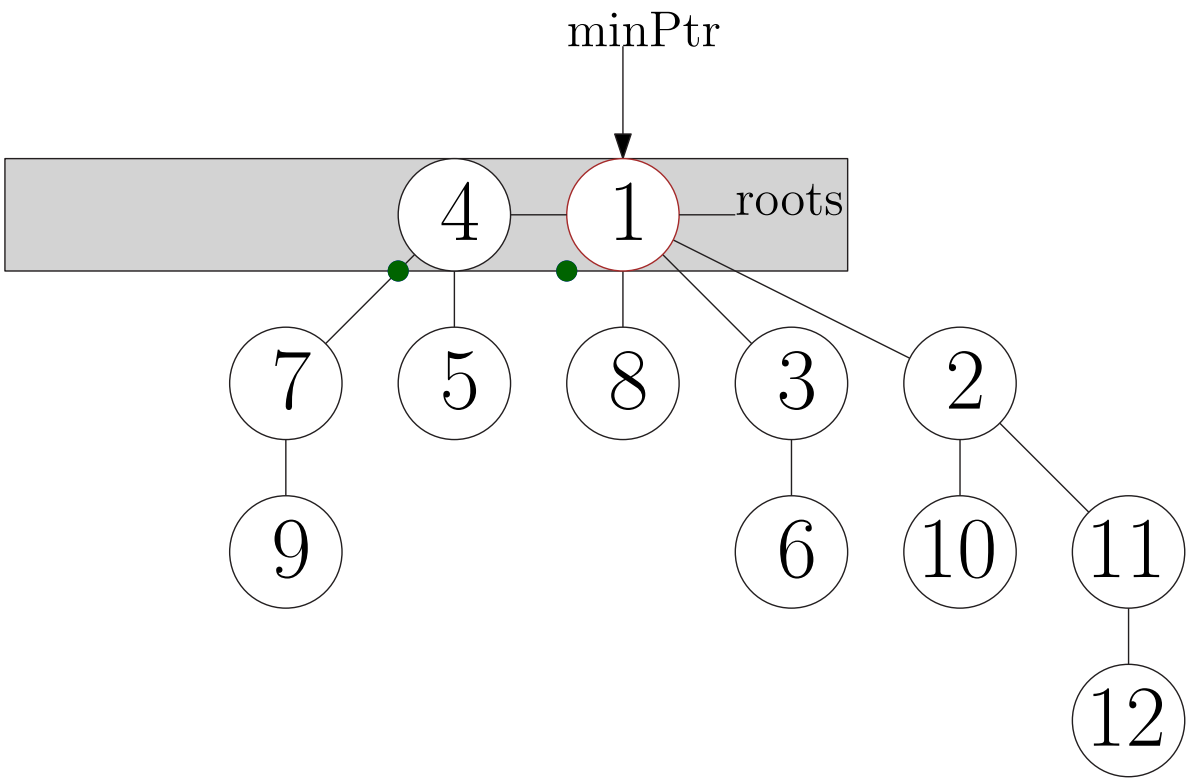
Fibonacci Heaps - Delete Min



Fibonacci Heaps - Delete Min



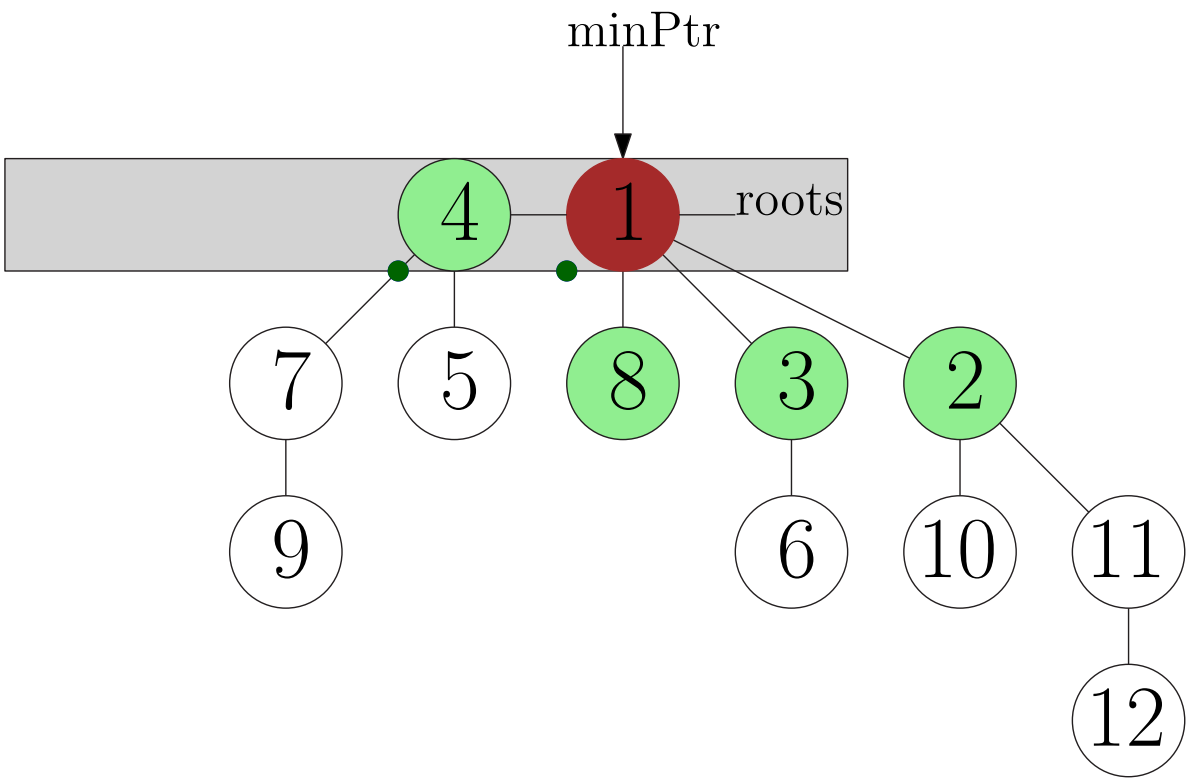
Fibonacci Heaps - Delete Min



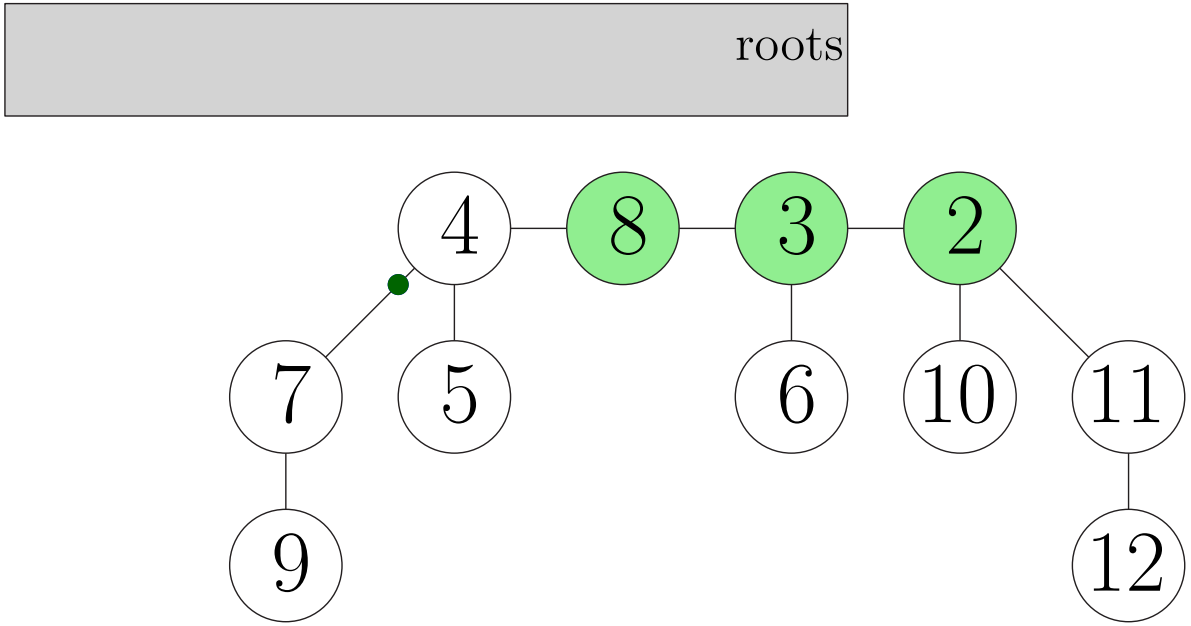
Fibonacci Heaps - Delete Min

- Union nach **Rang** $\rightarrow$ logarithmischer maxRank
- Vergleich: Pairing Heaps **streng paarweise**
- Rang bezieht sich nur auf **direkte** Nachfolger
- Frage: Wie lange kann eine einzelne DeleteMin Operation laufen?

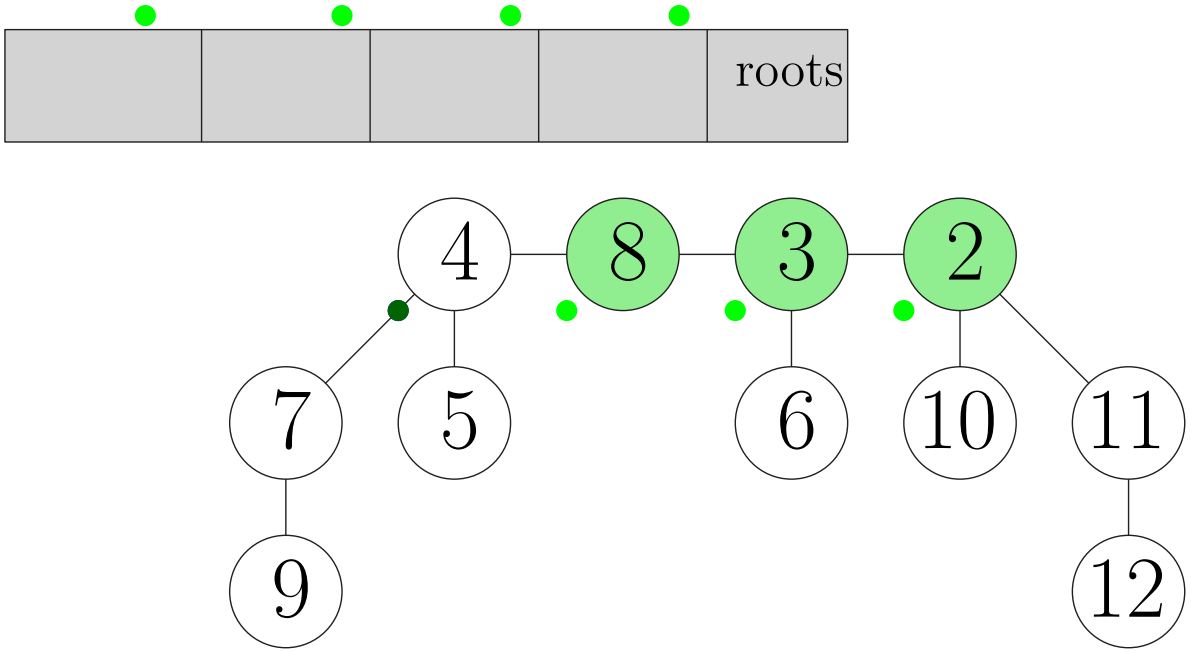
Fibonacci Heaps - Delete Min



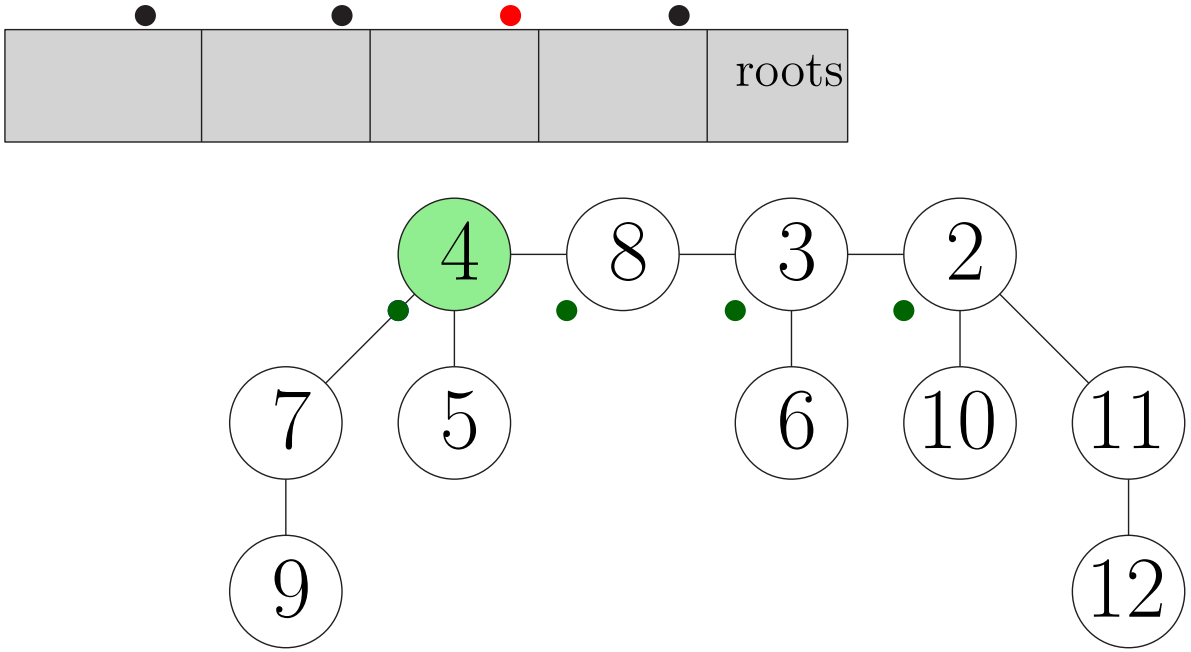
Fibonacci Heaps - Delete Min



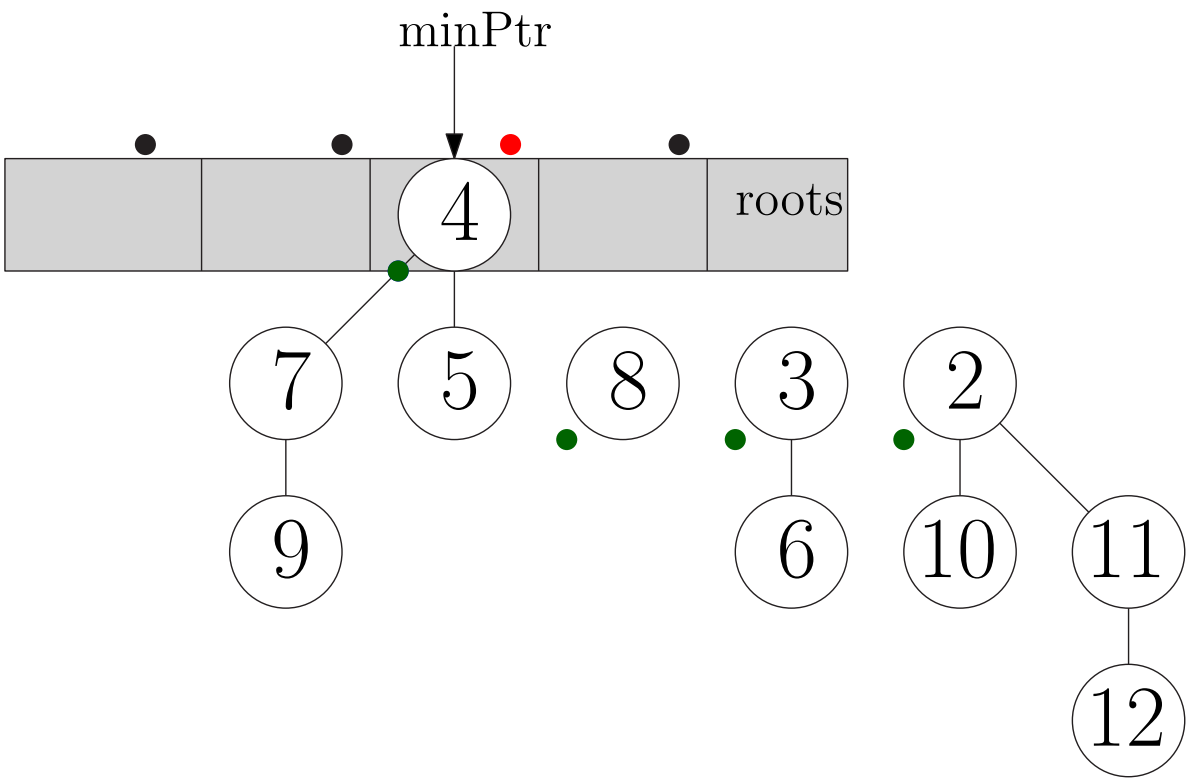
Fibonacci Heaps - Delete Min



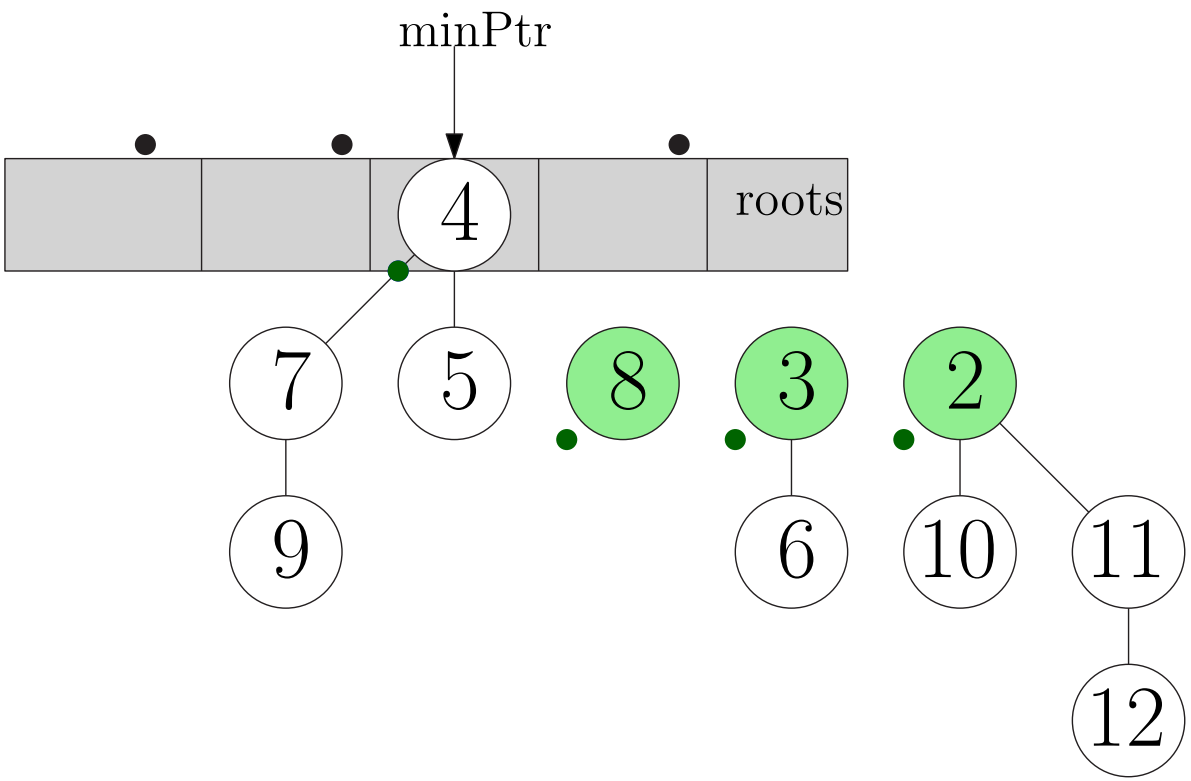
Fibonacci Heaps - Delete Min



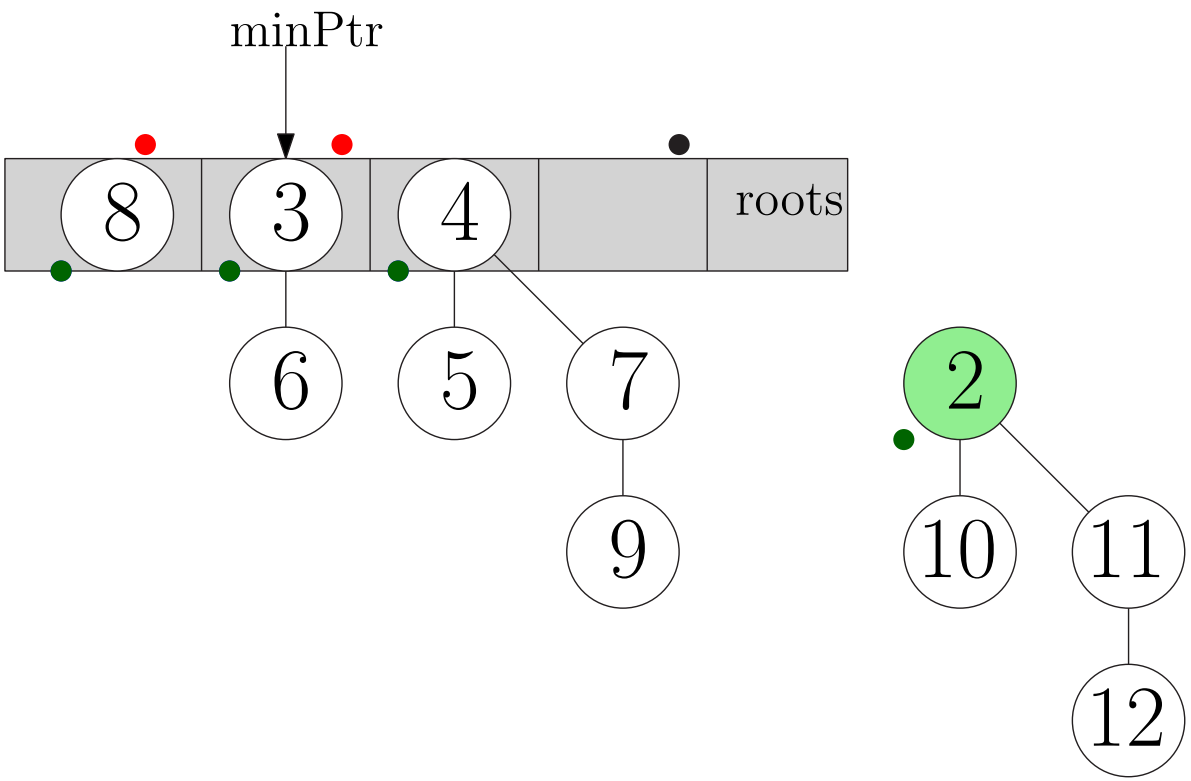
Fibonacci Heaps - Delete Min



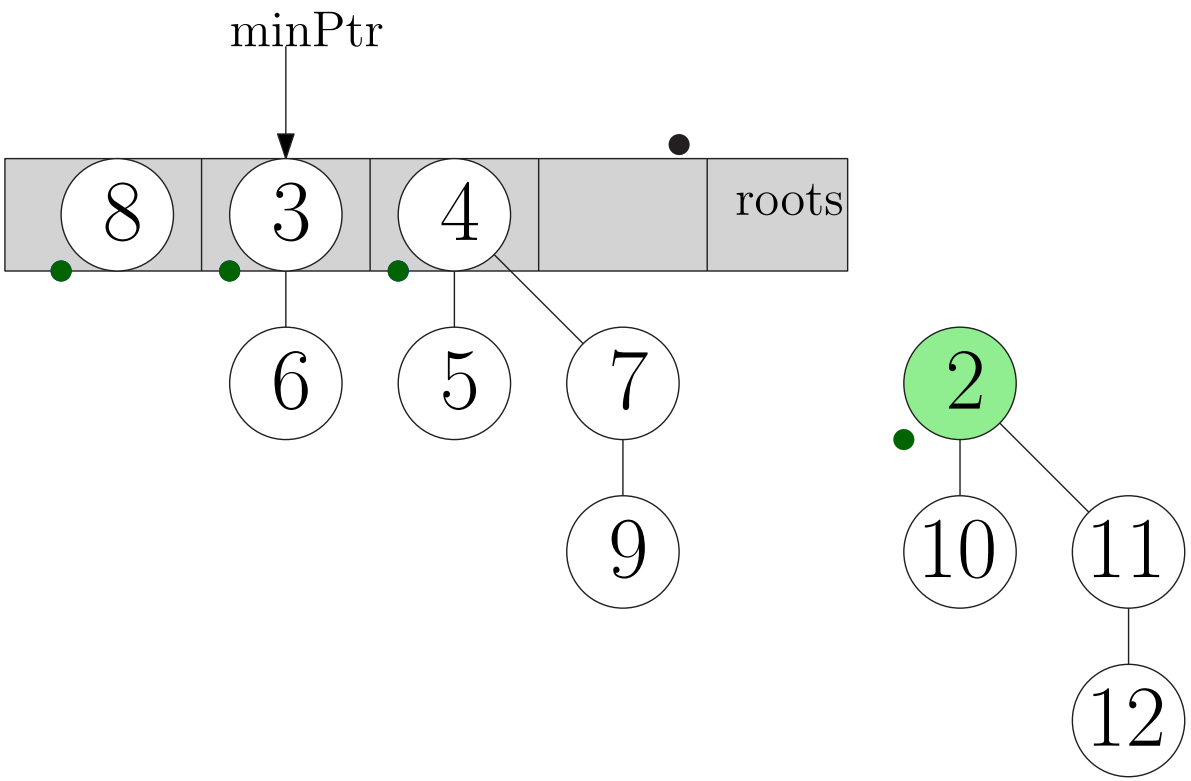
Fibonacci Heaps - Delete Min



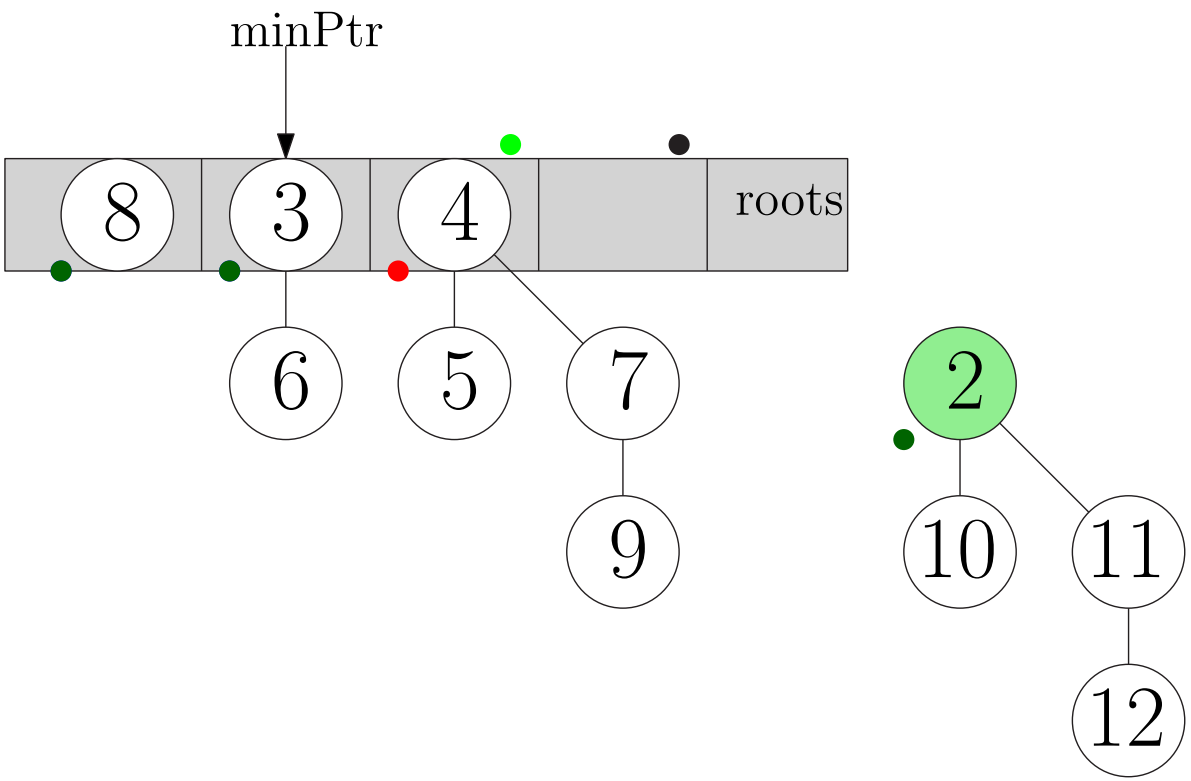
Fibonacci Heaps - Delete Min



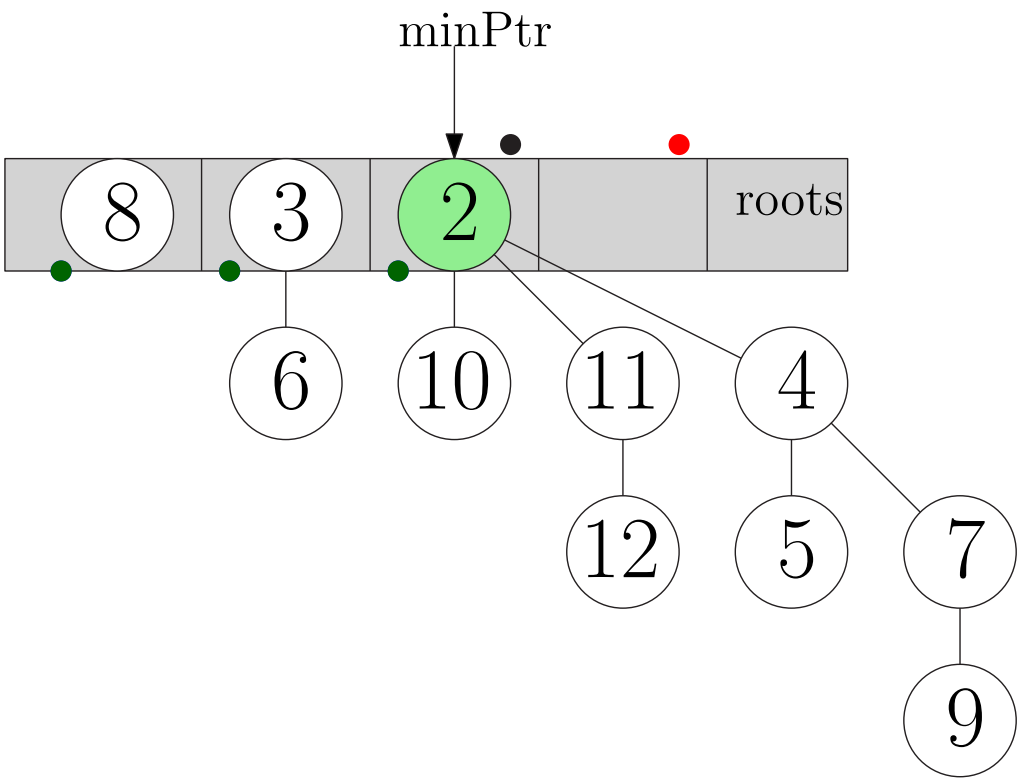
Fibonacci Heaps - Delete Min



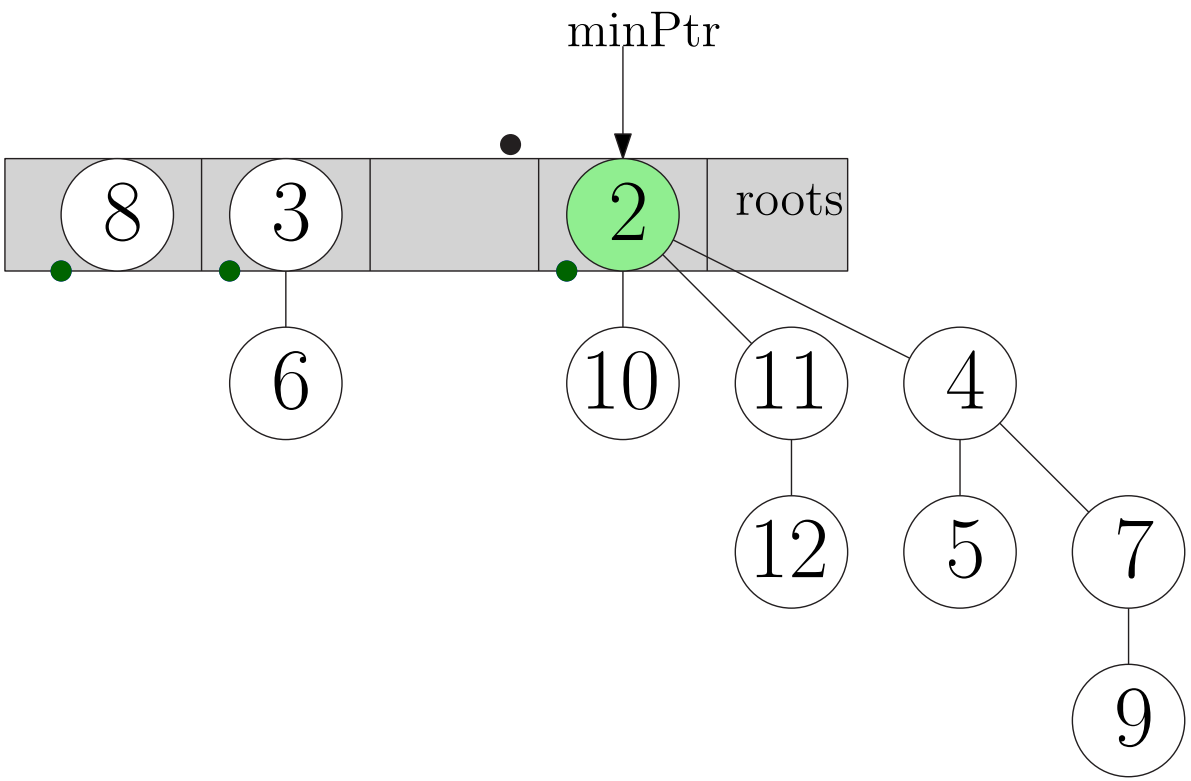
Fibonacci Heaps - Delete Min



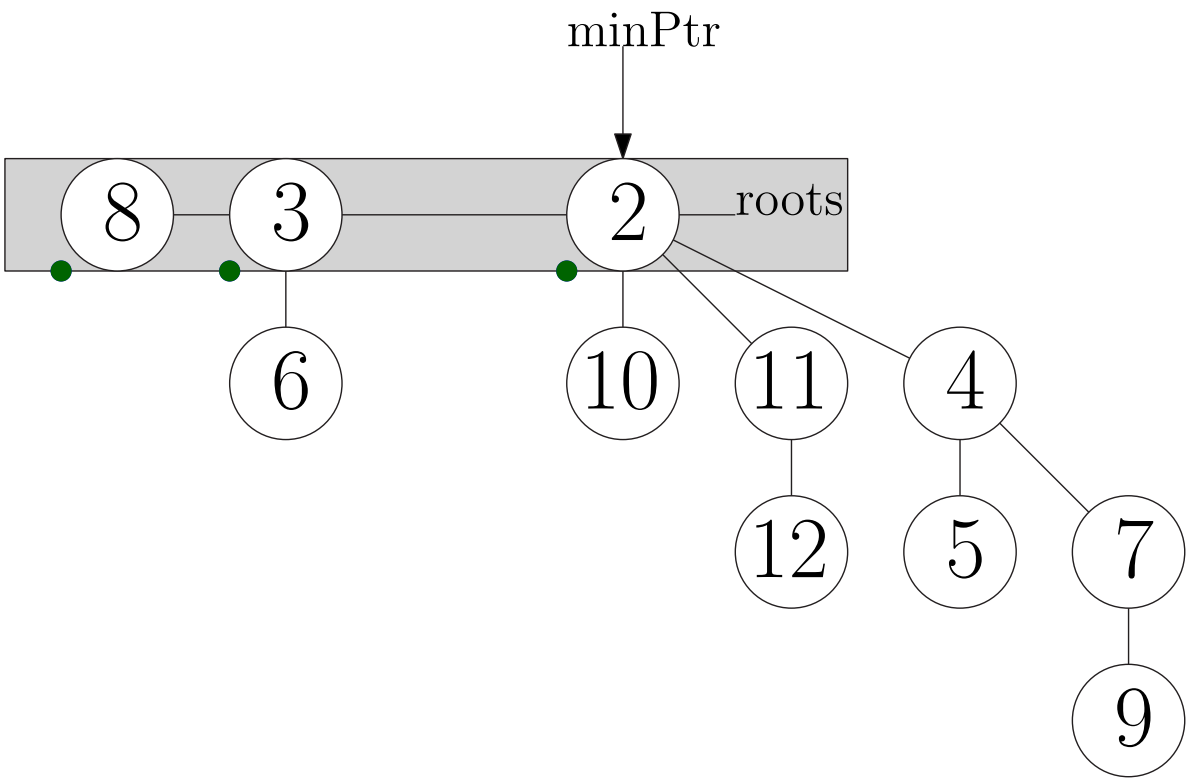
Fibonacci Heaps - Delete Min



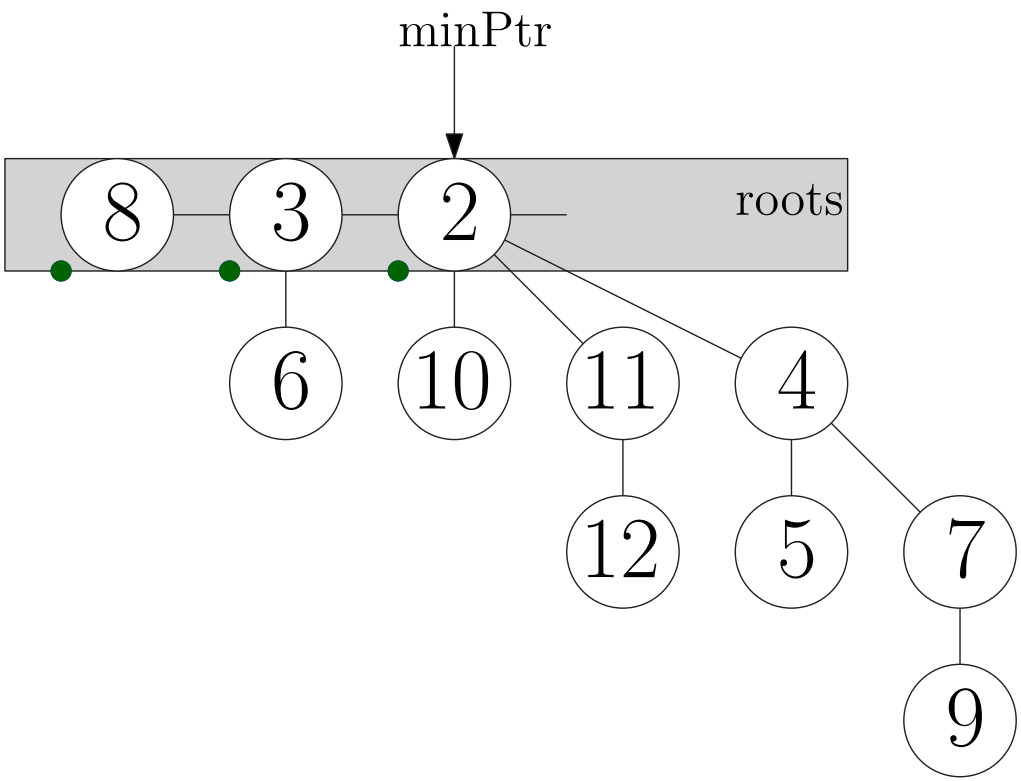
Fibonacci Heaps - Delete Min



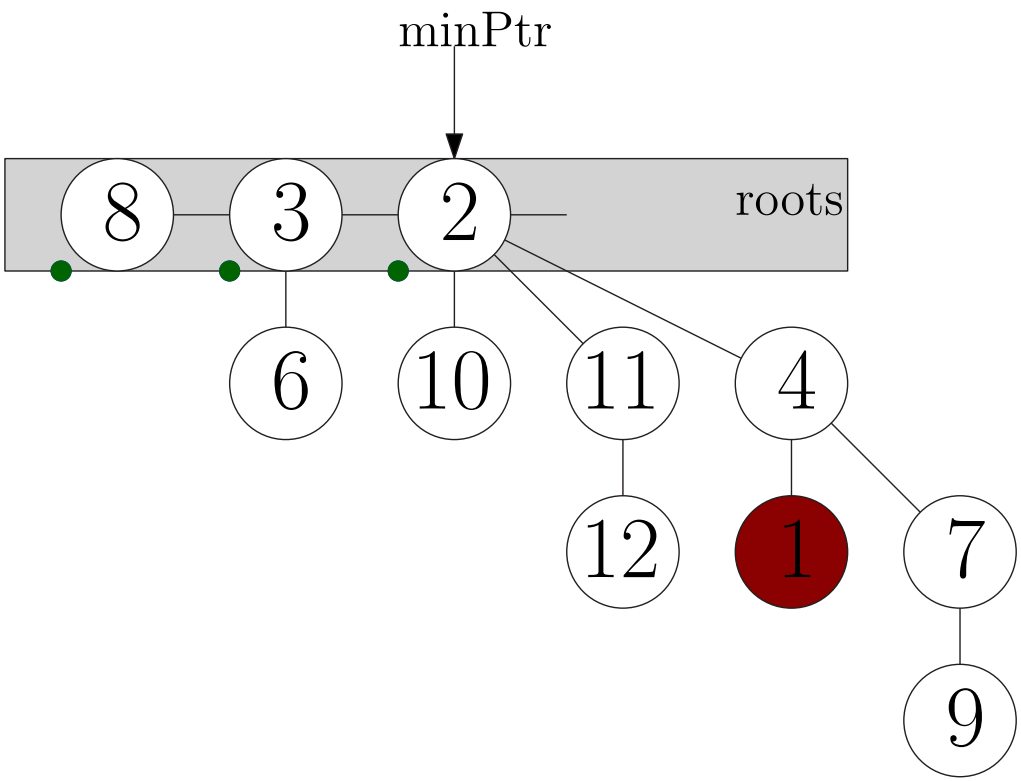
Fibonacci Heaps - Delete Min



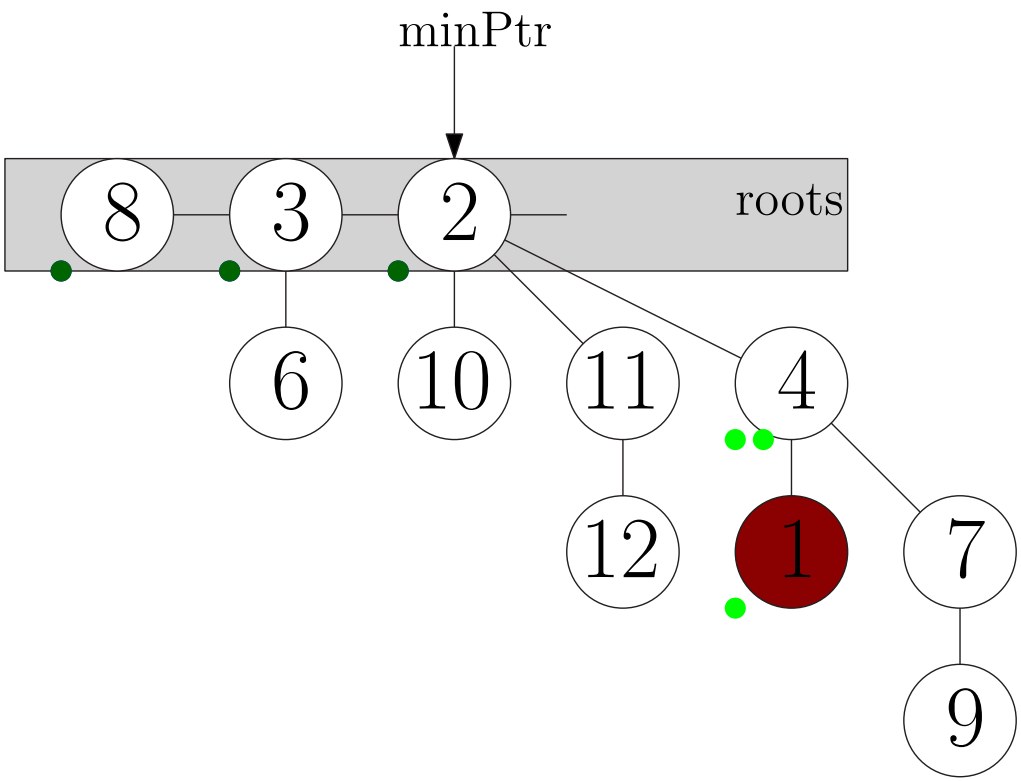
Fibonacci Heaps - Delete Min



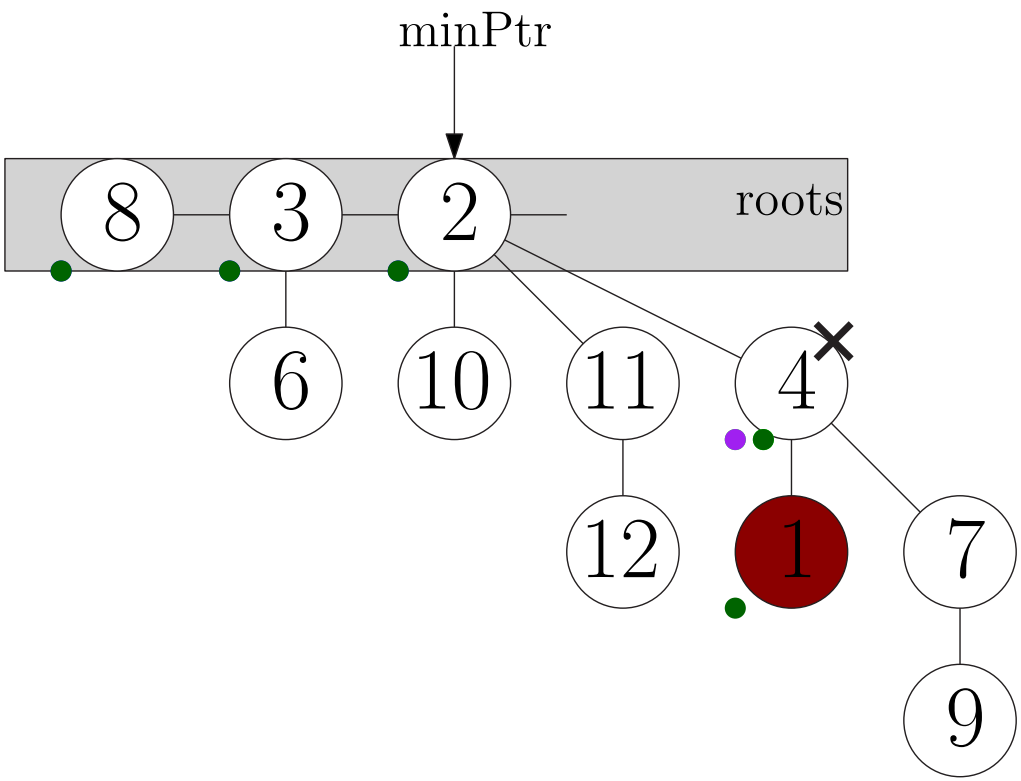
Fibonacci Heaps - Decrease Key



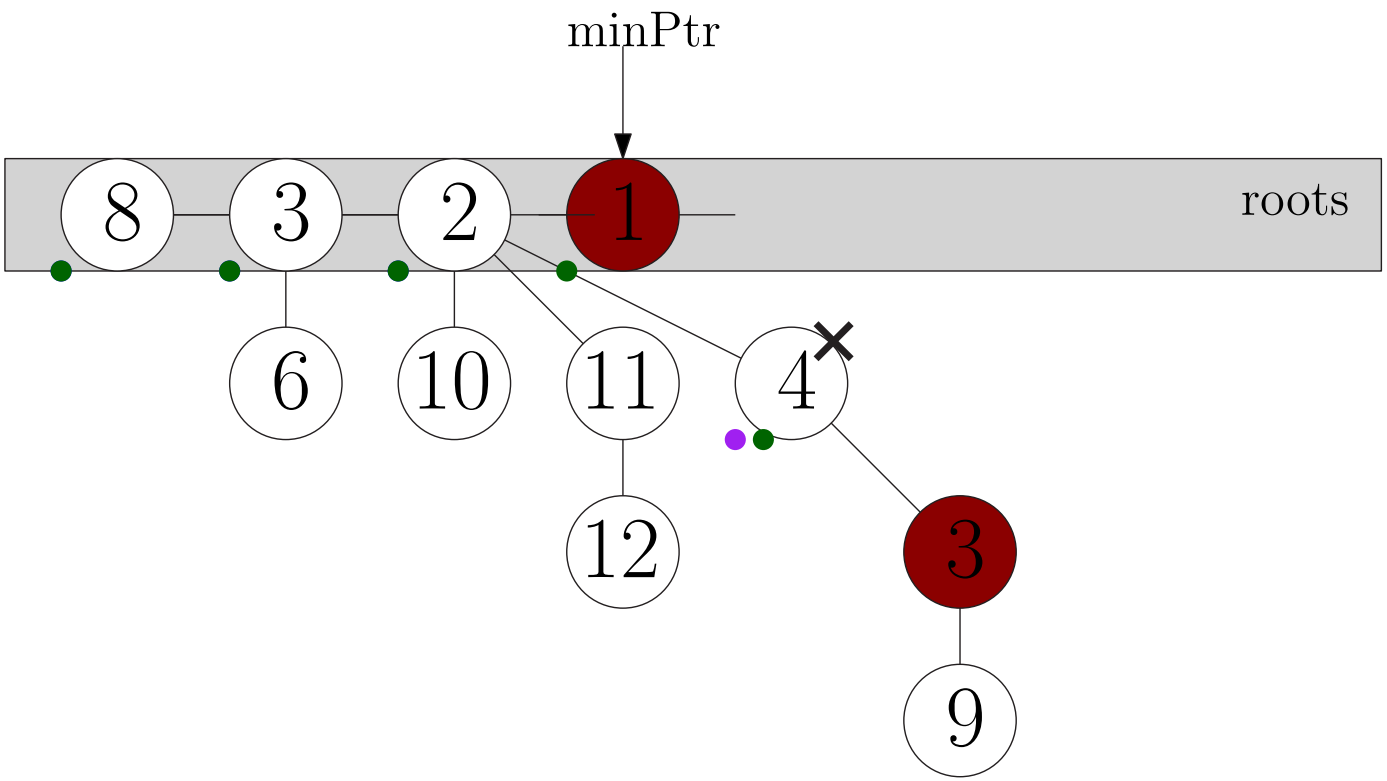
Fibonacci Heaps - Decrease Key



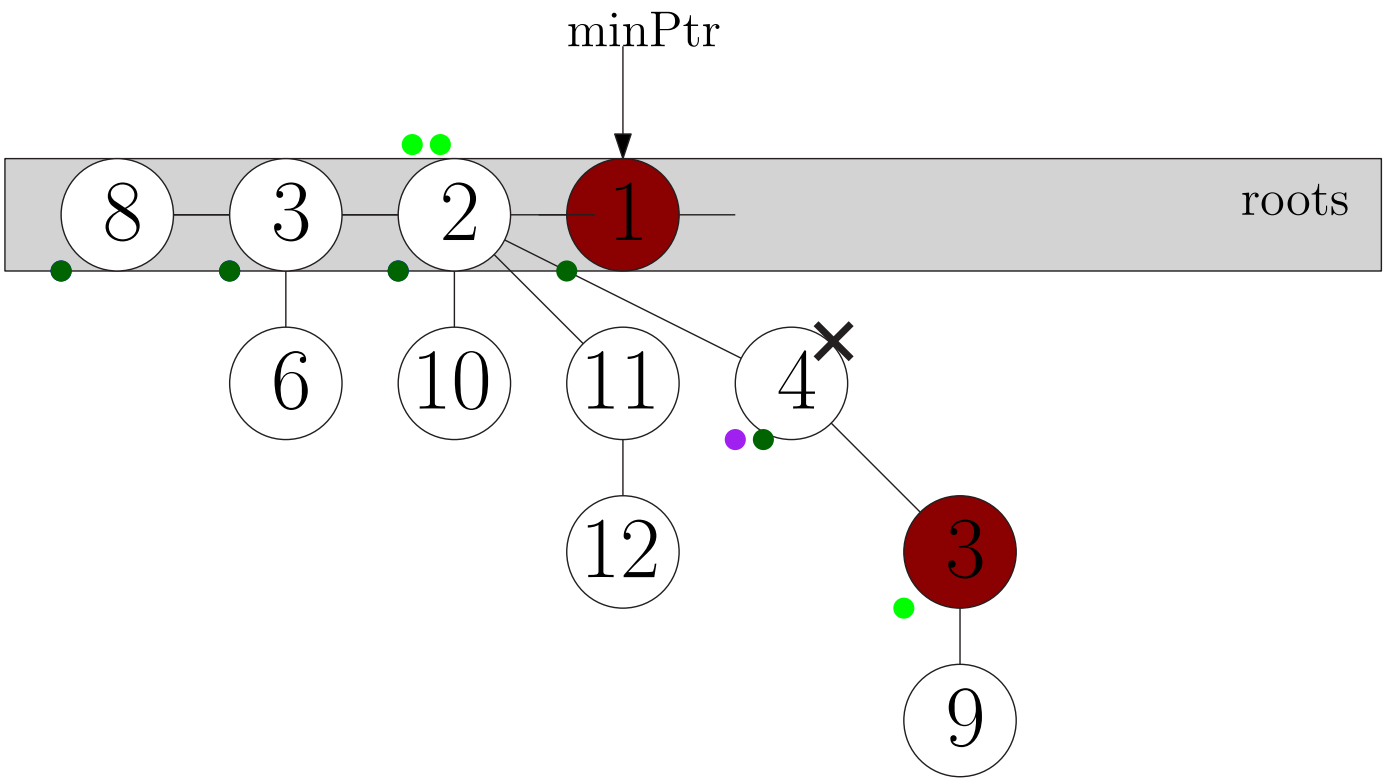
Fibonacci Heaps - Decrease Key



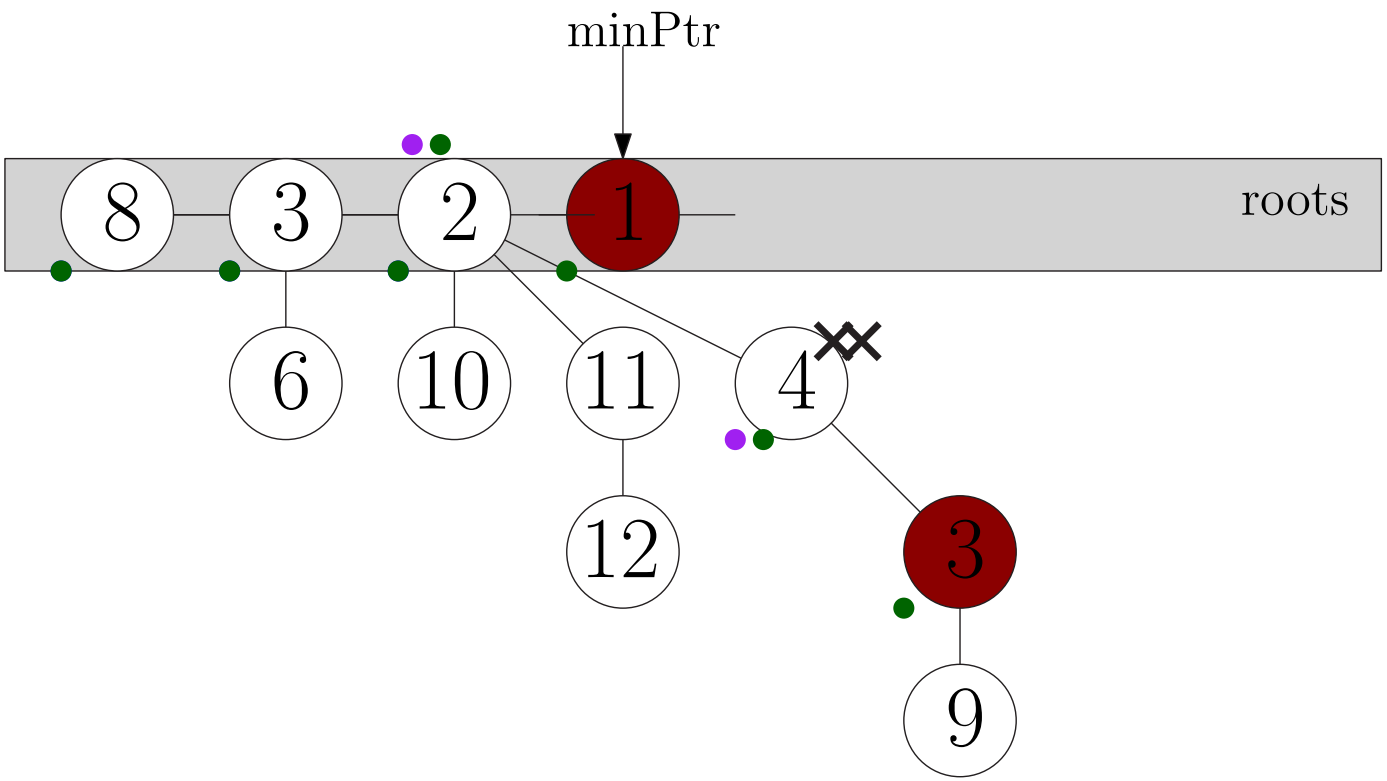
Fibonacci Heaps - Decrease Key



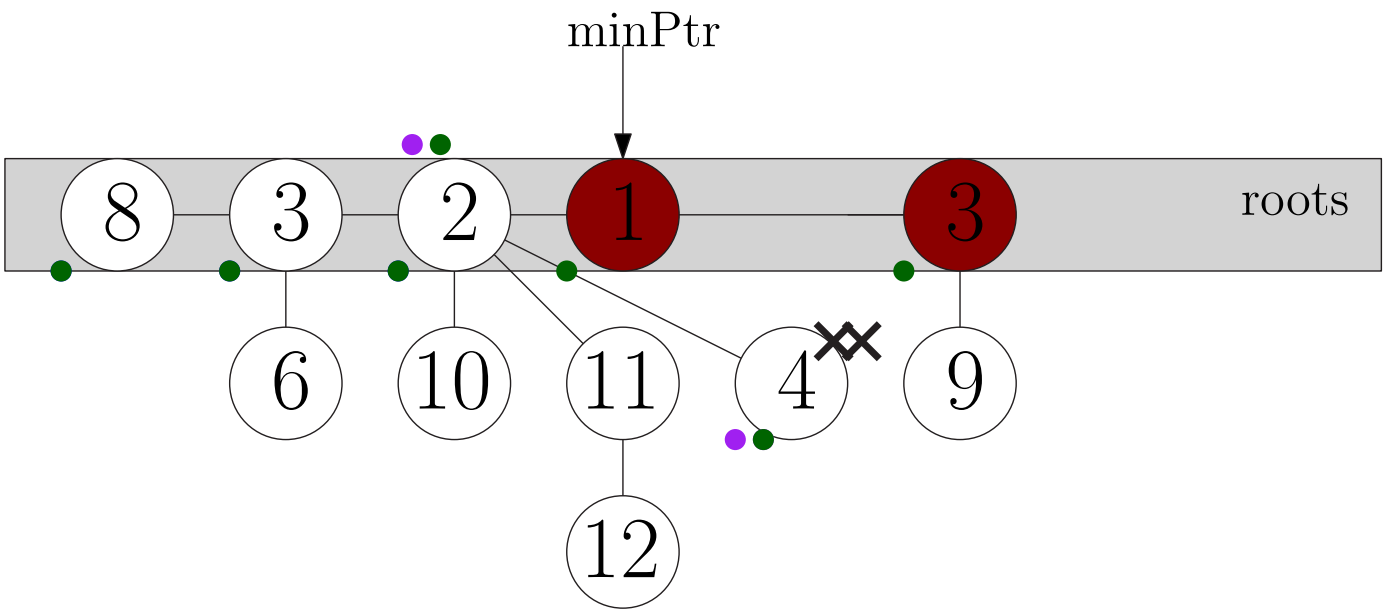
Fibonacci Heaps - Decrease Key



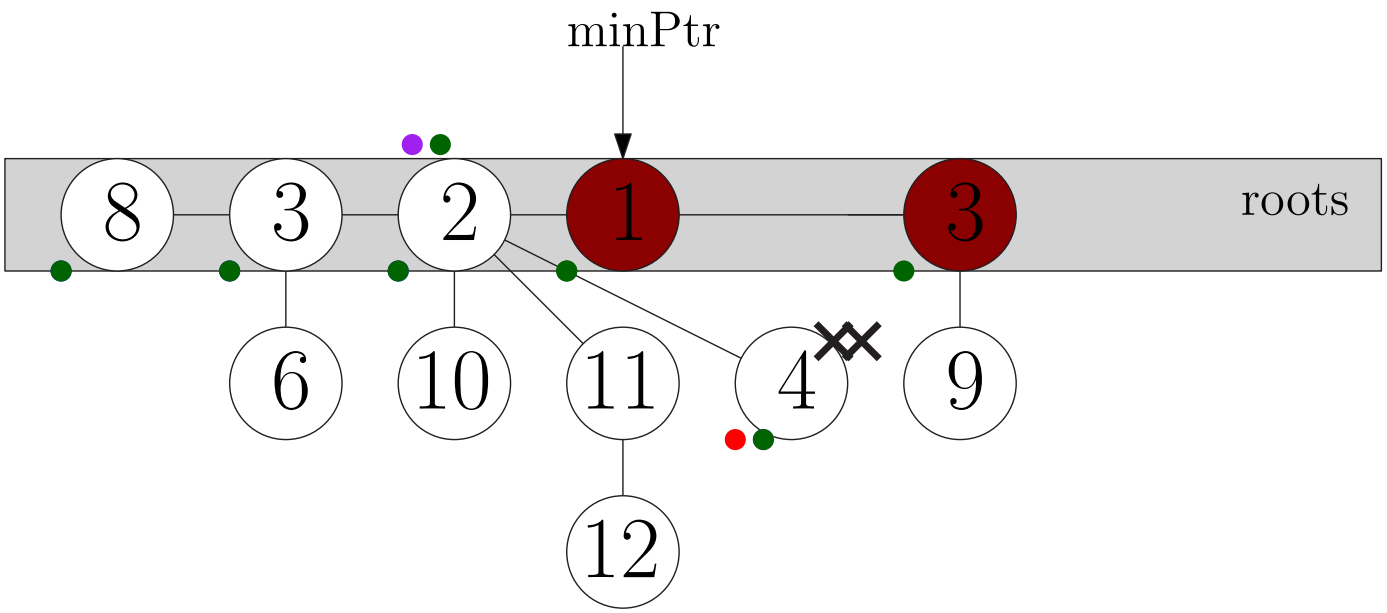
Fibonacci Heaps - Decrease Key



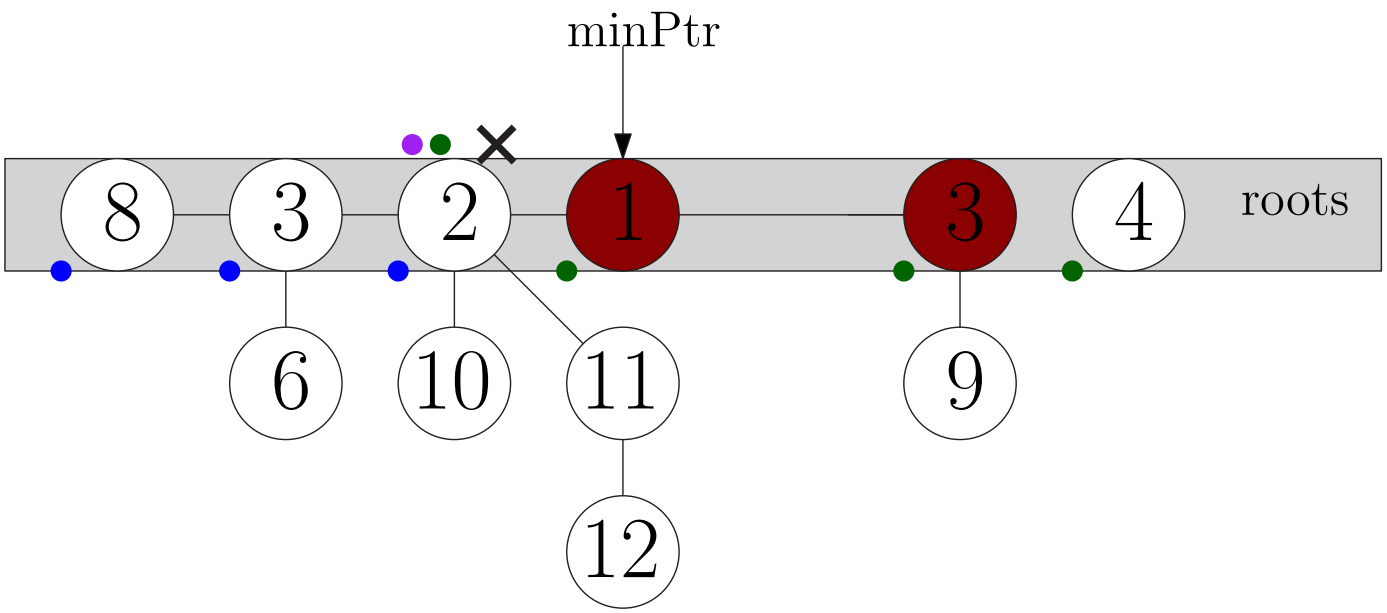
Fibonacci Heaps - Decrease Key



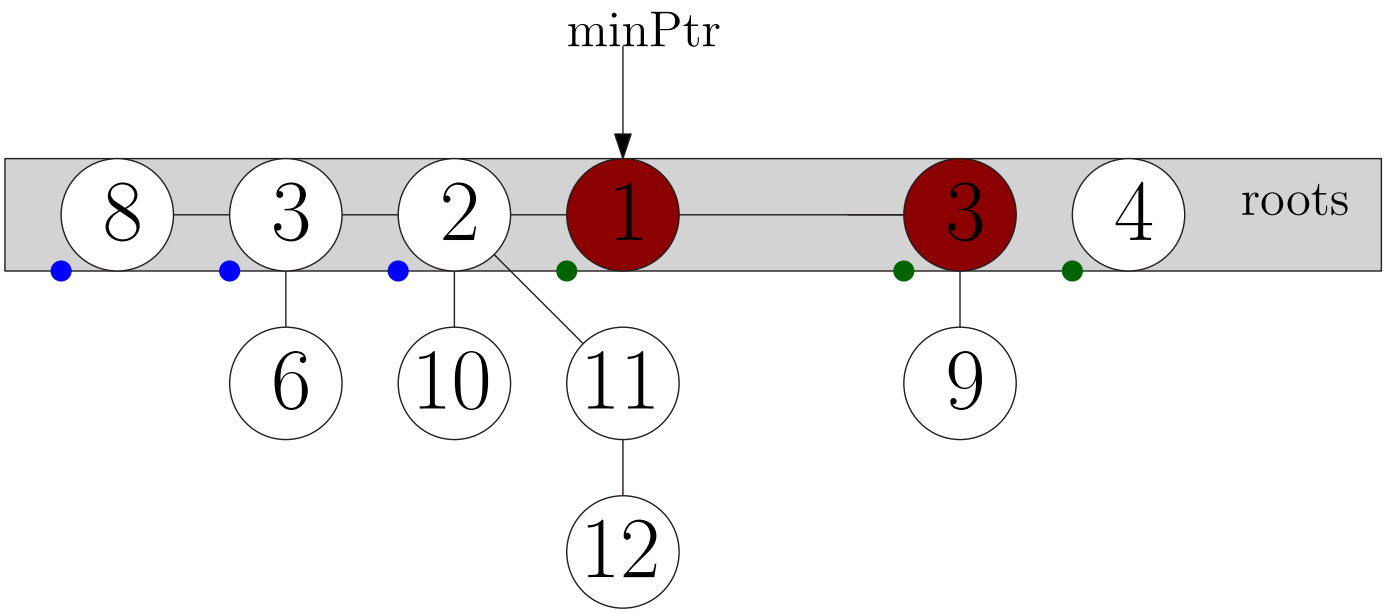
Fibonacci Heaps - Decrease Key



Fibonacci Heaps - Decrease Key



Fibonacci Heaps - Decrease Key



Fibonacci Heaps

- Amortisierte Laufzeiten sind **optimal** (vergleichsbasiert)
- **Einzelne Operationen** Kosten aber deutlich **länger**
- Ausprobieren, z.B. unter http://www.cse.yorku.ca/∼aaw/Jason/FibonacciHeapAnimation.html

Ende!

Feierabend!

Legende

Karlsruher Institut für Technologie

- ✕ Markierung - Kennzeichnet Knoten, die ein Kind verloren haben
- ✕✕ Doppelte Markierung - Temporäre Markierung. Muss sofort aufgelöst werden
- ● Neues, unbenutztes Token
- ● InsertTree-Token - kann genau 1 InsertTree Operation bezahlen
- ● Bezahltoken - wird zum Bezahlen genutzt
- ● Union-Token - zahlt für 1 Union Operation; zusätzlich einen InsertTree-Token erstellen
- ● Mark Token - Zahlt Cut und Markierung des Parents
- ● ● Aktive Knoten
- ● Knoten zum Löschen oder Cutten