

# Breitensuche

Lösung des **einfachen**  
Rucksackproblems mit Breitensuche

Breitensuche = Breite zuerst

# Breitensuche

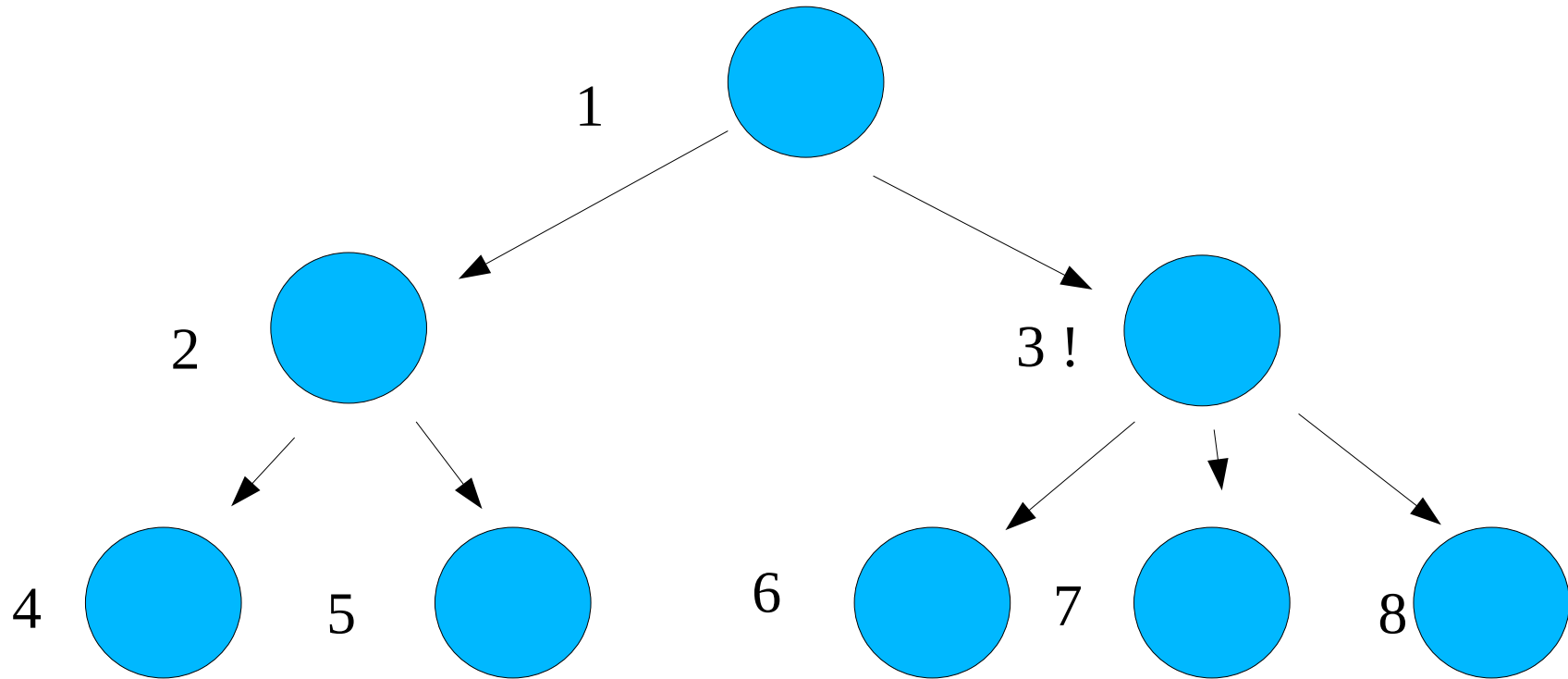
Breitensuche = Breite zuerst

Das heißt:

Die zuletzt bearbeitete Alternative wird nicht wie bei der Tiefensuche zuerst weiter bearbeitet, sondern zuletzt!

Also müssen die weiteren noch zu bearbeitenden Alternativen bis zur Bearbeitung gesichert werden.

# Breitensuche



# Breitensuche

Es ist egal, ob der Knoten 3 gleich bearbeitet wird,  
aber:

Werden die Nachfolger vom Knoten 2 generiert,  
muss das Programm sich noch merken, dass auch  
noch die Nachfolger vom Knoten 3 zu generieren  
sind, **bevor** einer der Knoten 4 oder 5 bearbeitet  
wird.

4 und 5 müssen also „hinten ran“.

# Breitensuche

Eine Breitensuche ohne **Warteschlange (queue)** geht nicht!

Beim Aufruf ist diese Warteschlange leer und wird bei der Bearbeitung jeweils **vorn geleert** und **am Ende gefüllt**.

Alle Alternativen einer Ebene des Expansionsgraphen werden also **hinten** angehängt.

# Breitensuche

Ein Problem beim Rucksackproblem ist:  
Wie merke ich, dass ich schon alle Alternativen  
einer Ebene versucht habe?

Ich muss nämlich beim Wechsel in die nächste  
Ebene ein neues Stück verwenden.

# Breitensuche

Wie merke ich, dass ich schon alle Alternativen einer Ebene versucht habe?

Eine mögliche Lösung:

Dazu wird für die Bearbeitung einer Ebene zusätzlich ein akku eingeführt

In den wird die gerade bearbeitete Alternative getan.

Nachdem eine Ebene fertig bearbeitet ist, wird der akku komplett in queue übernommen.

Eine andere – vielleicht einfachere- Lösung:

Es werden durch eine weitere Funktion alle Alternativen der Ebene sofort generiert.

# Breitensuche

Die iterative Version der Breitensuche benötigt als Parameter

- **die Stückeliste**
- **den Zielwert** ( Fassungsvermögen des Containers)

und als interne Variable

- **die queue** (Alternativen)
- **den akku** (Alternativen der nächsten Ebene)



# Breitensuche

Auch bei Iteration kläre:

Welche Möglichkeiten gibt es in den Schritten / den Stufen ?

- Alle Stücke bearbeitet (for-Schleife einsetzen)
- Zielwert erreicht ( → Ausgabe!)
- Zielwert überschritten (weglassen)
- akku leer (bedeutungslos)
- queue leer (akku in die queue für die nächste Ebene)
- else: Standardfall

# Breitensuche

Das war's.

*[Wie immer: Nicht **die** Lösung sondern **eine**!]*