



Enabling the Creation
of Intelligent Things



European Driving License for Robots and Intelligent Systems

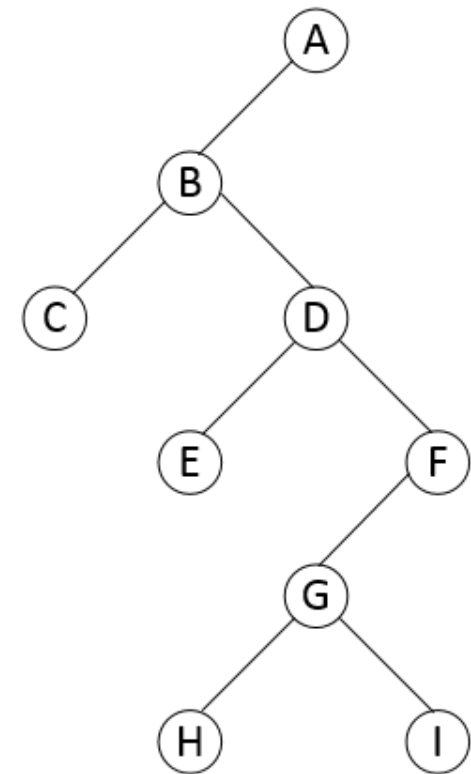
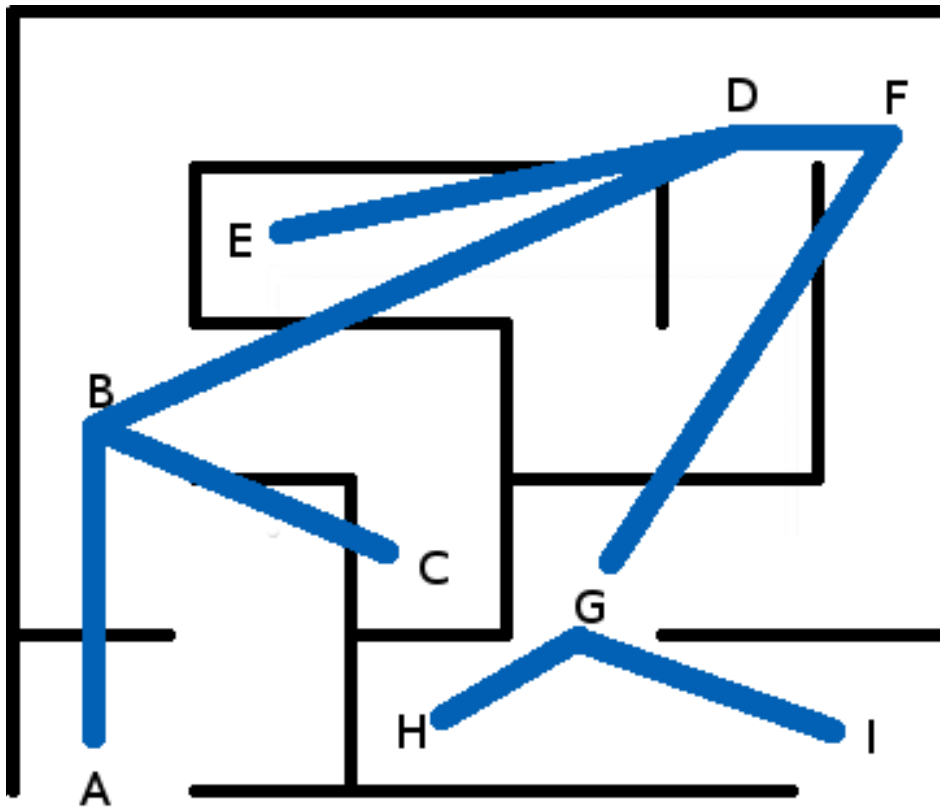


Problemlösung durch Suche



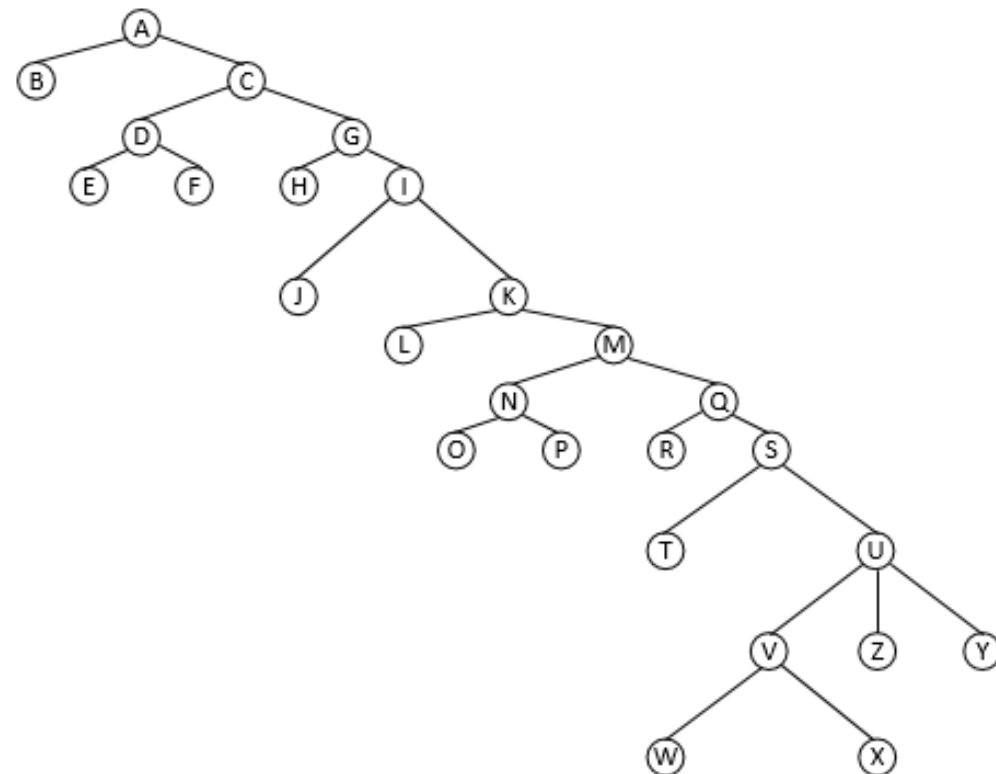
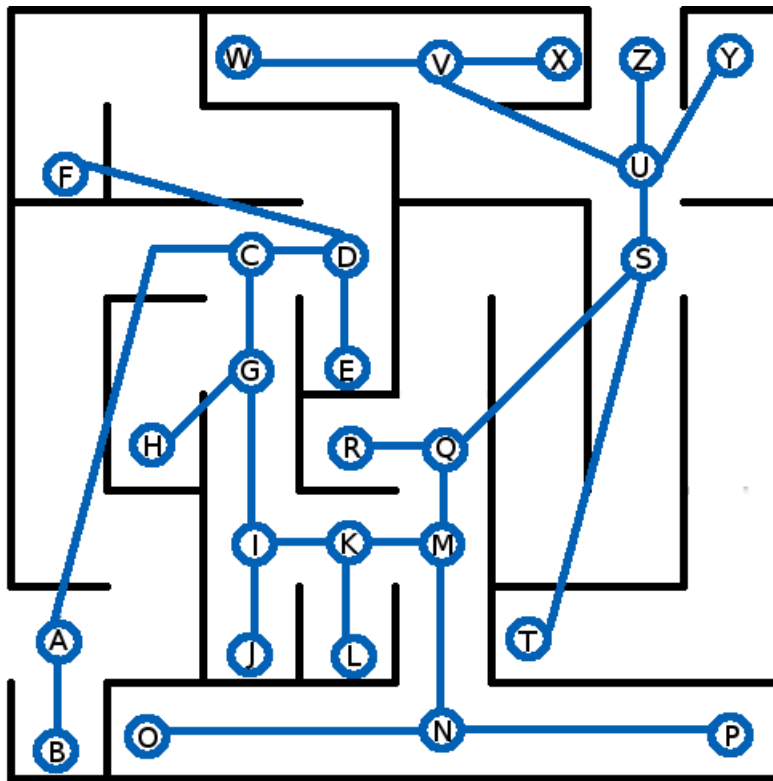
Actives Plenum

Zeichne das Labyrinth als Graph und als Baum

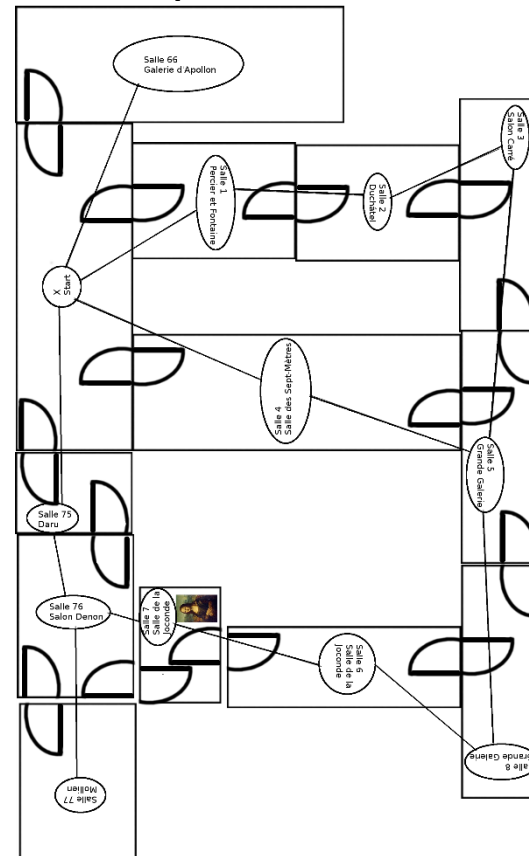
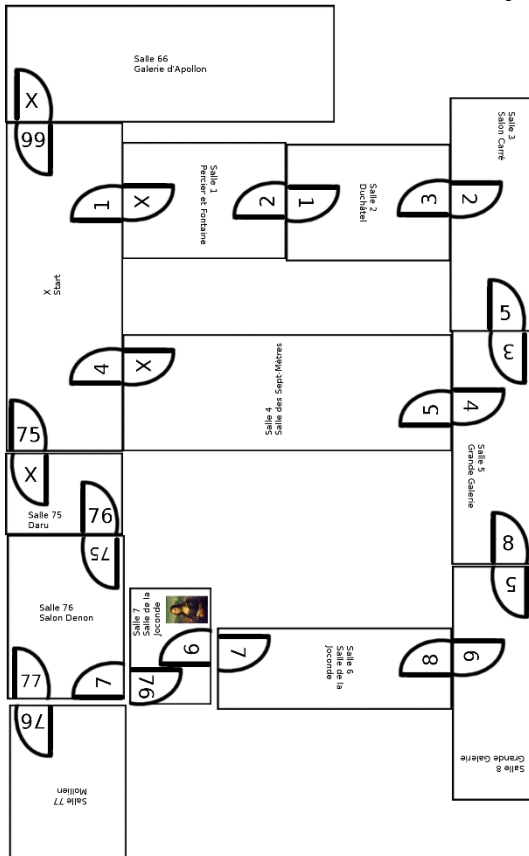


Actives Plenum

Zeichne das Labyrinth als Graph und als Baum

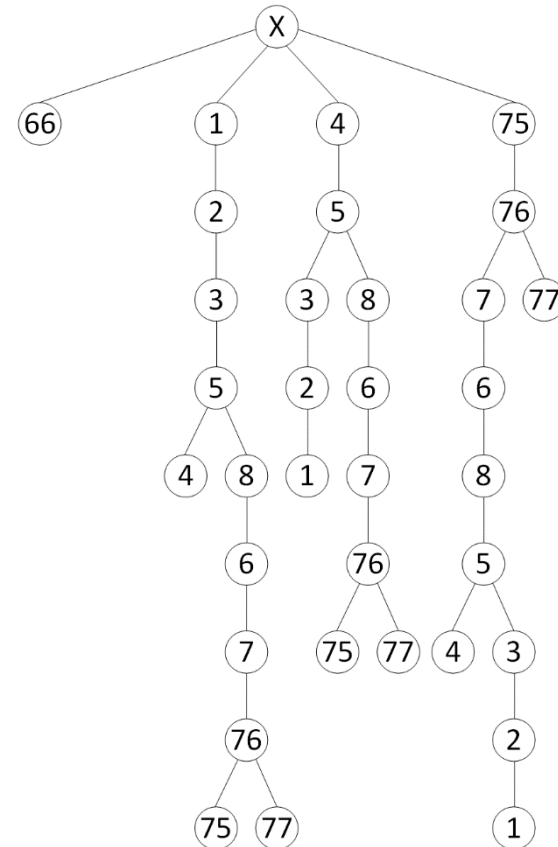
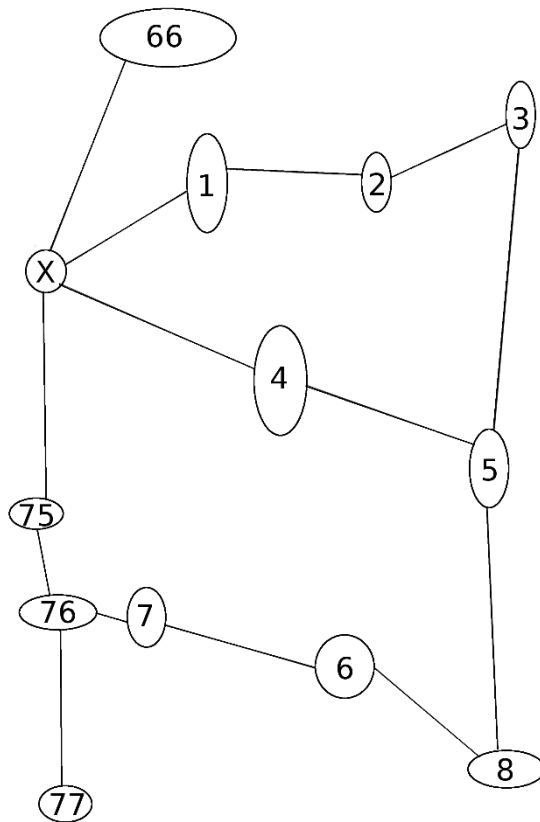


Zu Zweit Zeichnet das Labyrinth als Graph und als Baum



Lösung Mona Lisa

Zeichnet das Labyrinth als Graph und als Baum



Der Faden der Ariadne

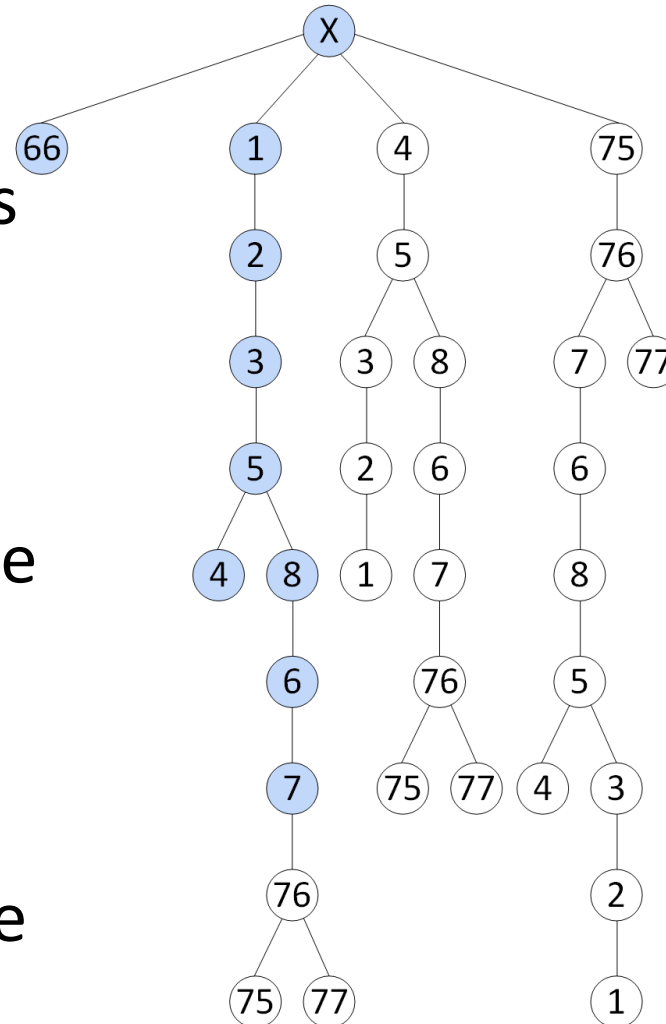


<https://www.youtube.com/watch?v=8qrZ1clEp-Y>

Tiefensuche

Gehe immer eine Ast des Baumes bis zum Ende.

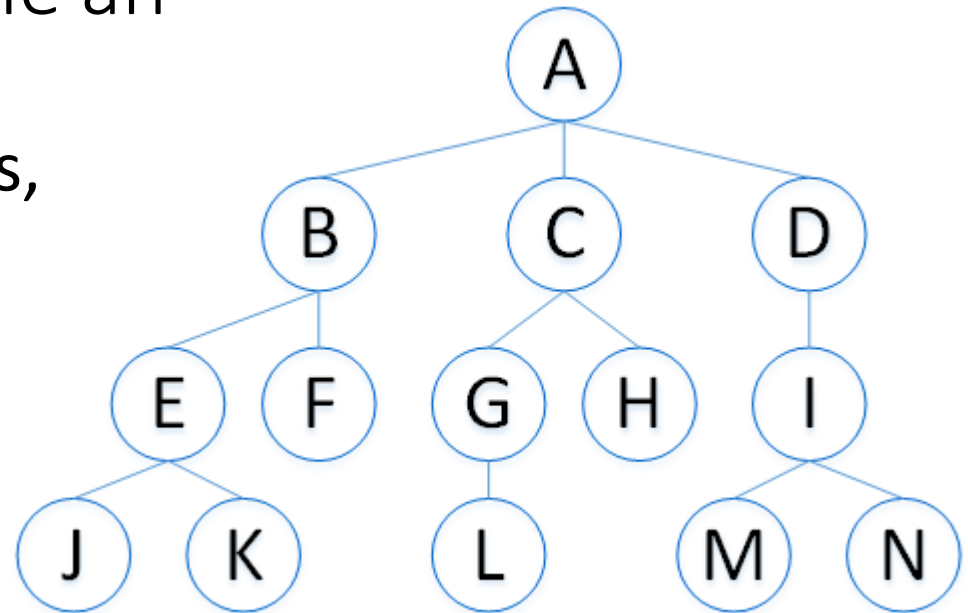
Wenn du an einen Endknoten kommst, ohne das Ziel gefunden zu haben, gehe zurück zum letzten Elternknoten mit Abzweigungen und suche dort weiter.



Wende die Tiefensuche an

Verwende die Stack-Cards,
um den Pfad anzugeben

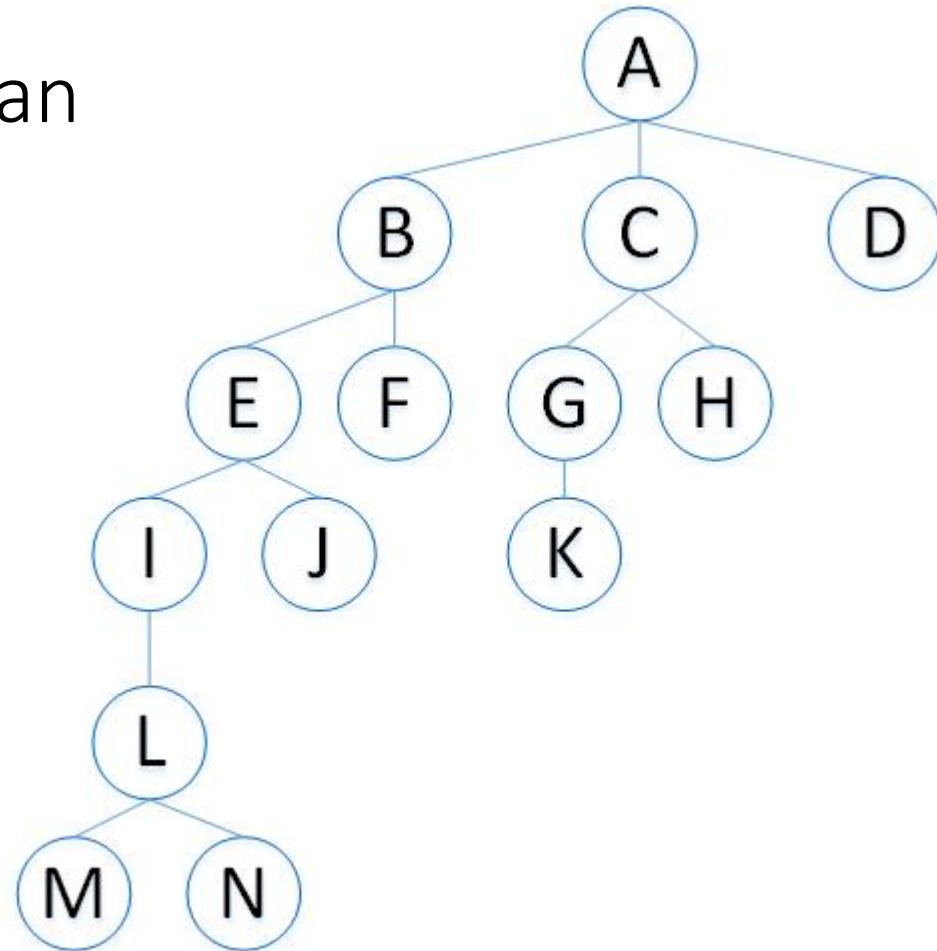
F ist der Zielknoten



Wende die Tiefensuche an

Verwende die Stack-Cards,
um den Pfad anzugeben

G ist der Zielknoten

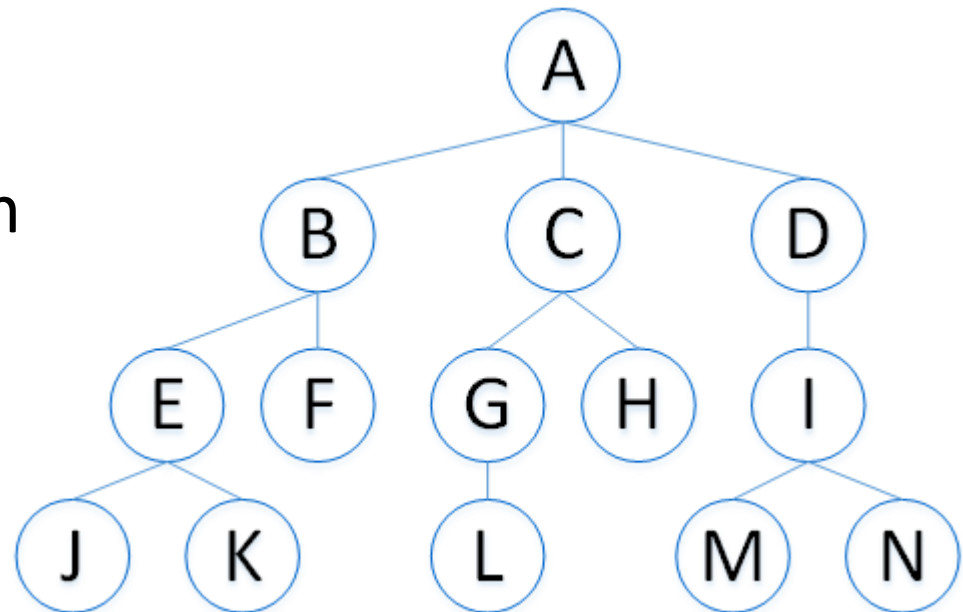


Was passiert, wenn du den Stack (Stapel) durch eine Queue (Schlange) ersetzt?

Verwende die Kärtchen, um die Schlange zu bilden

F ist der Zielknoten

Was hat sich verändert?

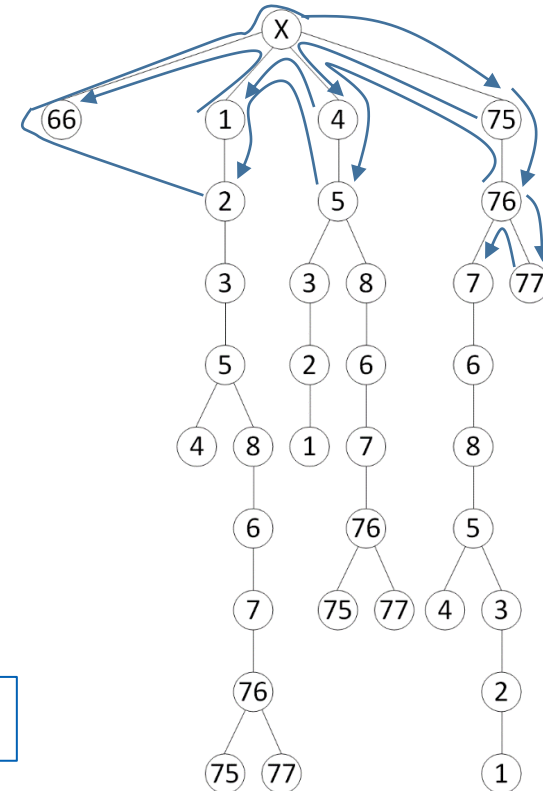


Breitensuche Algorithmus

- Initialisiere die Queue mit dem Anfangszustand
- Solange die Schlange nicht leer ist:
 - ➔ • Wenn der aktuelle Status ein Zielstatus ist
 - Erfolg
 - ➔ • Sonst, füge alle möglichen Wege zur Queue hinzu
Es ist egal in welche Richtung du zuerst gehst

Queue:

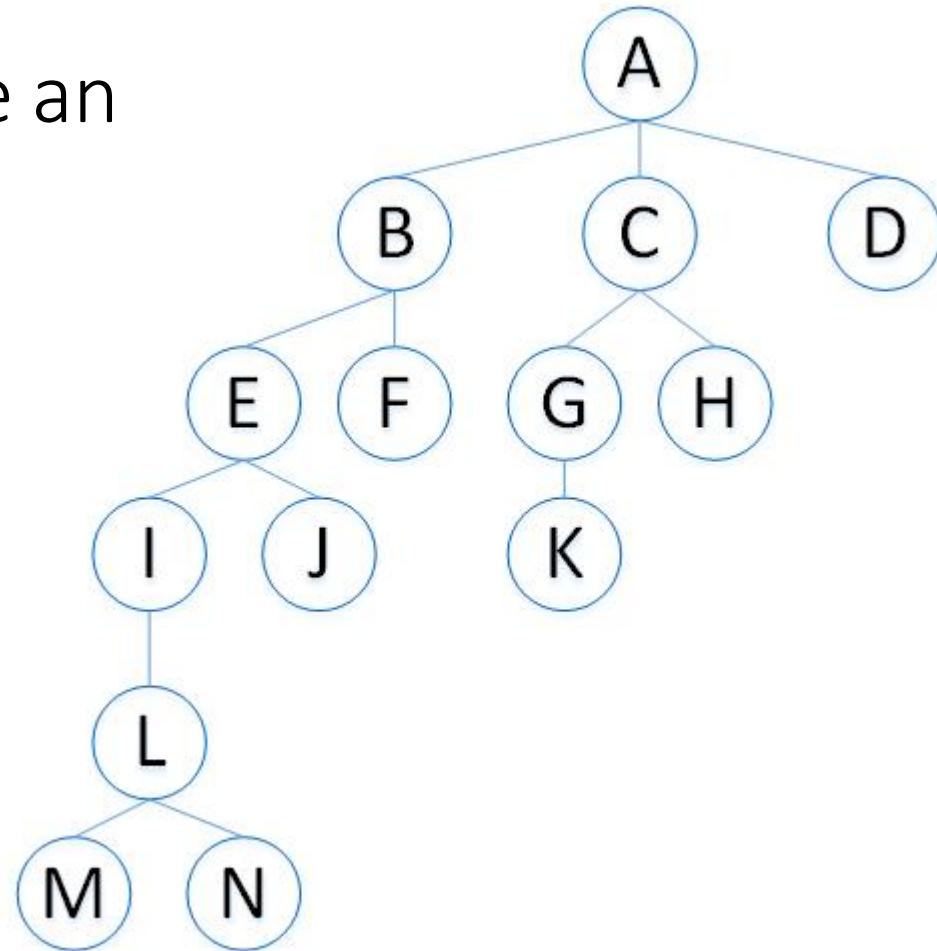
X	X,75	X, 4	X, 1	X, 66	X, 75, 76	X, 4, 5
X, 1, 2	X, 75, 76, 77	X, 75, 76, 7	X, 4, 5, 8			
X, 4, 5, 3	X, 1, 2, 3					



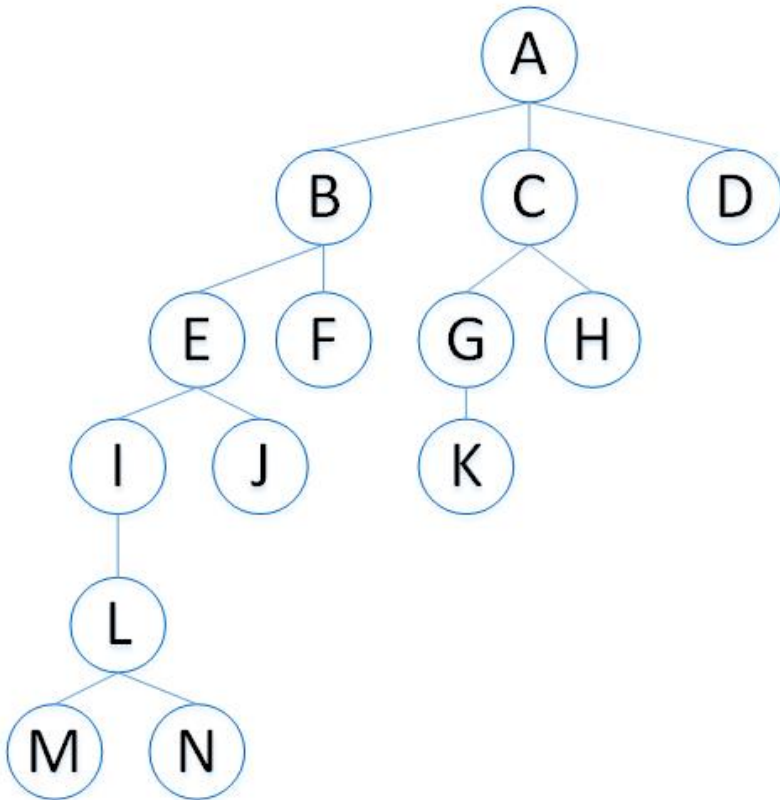
Wende die Breitensuche an

Verwende die Stack-Cards,
um den Pfad anzugeben

G ist der Zielknoten



Breitensuche vs. Tiefensuche



Breitensuche

Findet immer den dem Anfangsknoten am
nächstgelegenen Zielknoten

Wenn der Verzweigungsfaktor endlich ist
Wenn der dem Anfangsknoten
nächstgelegenen Zielknoten in endlicher
Tiefe liegt
ist der Suchalgorithmus vollständig (findet
immer eine Lösung)

Wenn alle Aktionen die gleichen Kosten
haben
Findet der Suchalgorithmus den kürzesten
Weg

Tiefensuche

Wenn der Suchbaum eine endliche Tiefe
hat
ist der Suchalgorithmus vollständig (findet
immer eine Lösung)



Priority Queue

- Sortiert die Elemente nach ihrer Priorität.
 - Kürzester Weg, geringste Kosten, höchste Priorität
- Ordne die Elemente nach Priorität:
 - 79245
 - Lösung: 24579

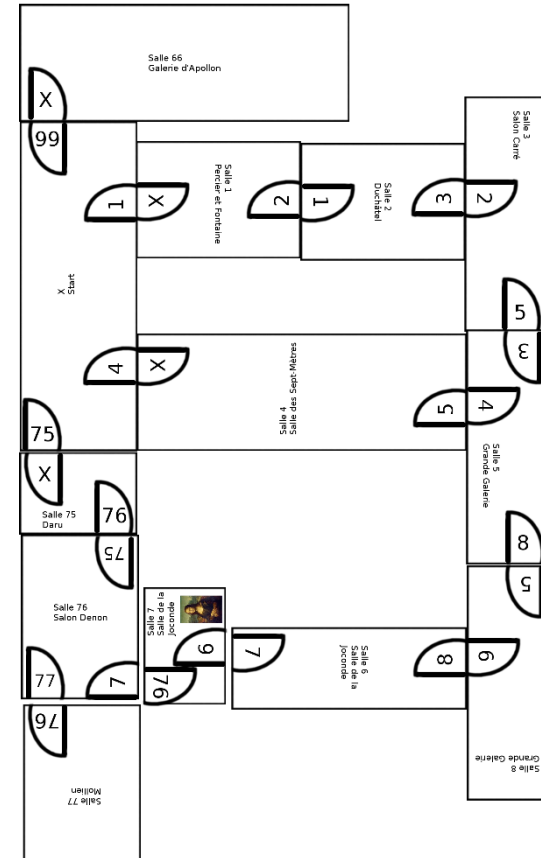
Priority Queue

- Sortiere die Queue, um sie in eine Priority Queue umzuwandeln:
(A,2), (B,7), (C,1), (D,4)
- Lösung:
(C,1), (A,2), (D,4), (B,7)

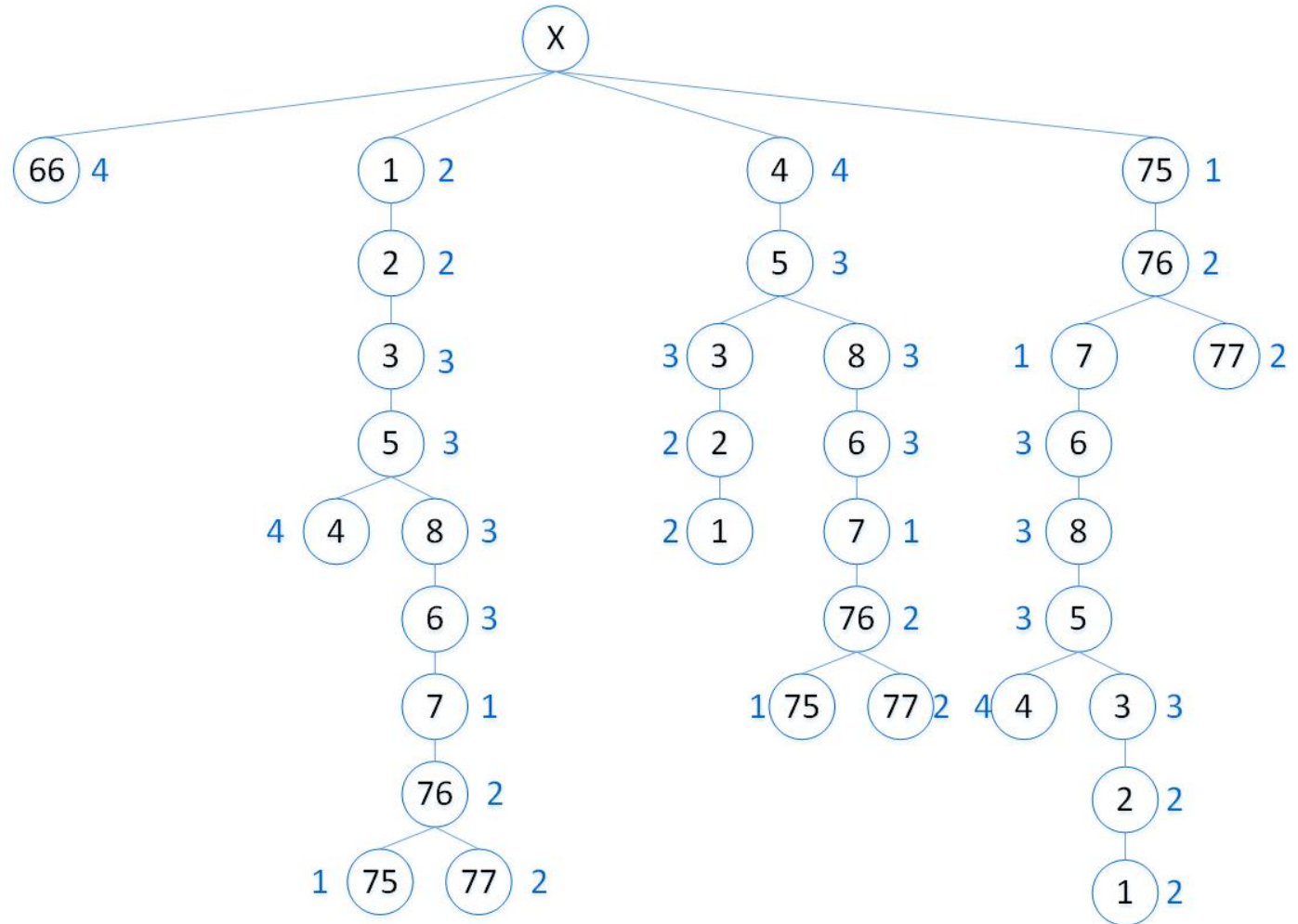
Prioritäten zu einem Suchbaum hinzufügen

- Ganz große Räume kosten 4 (z.B. 66)
- Große Räume kosten 3 (z.B. 5)
- Mittlere Räume kosten 2 (z.B. 77)
- Kleine Räume kosten 1 (z.B. 75)

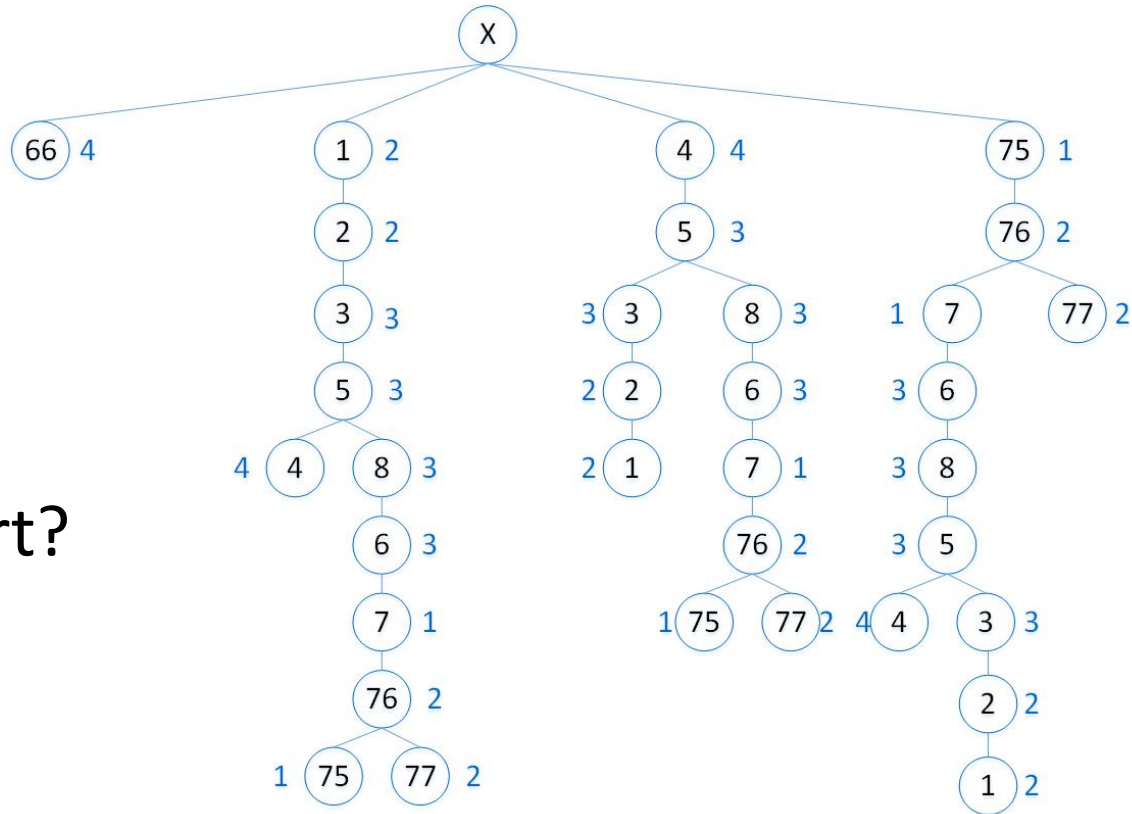
- Vermerke die Kosten am Suchbaum



Lösung



Was hat sich verändert?



Uniform Cost Search / Bestensuche (Knoten 7)

- Initialisiere die Queue mit dem Anfangszustand

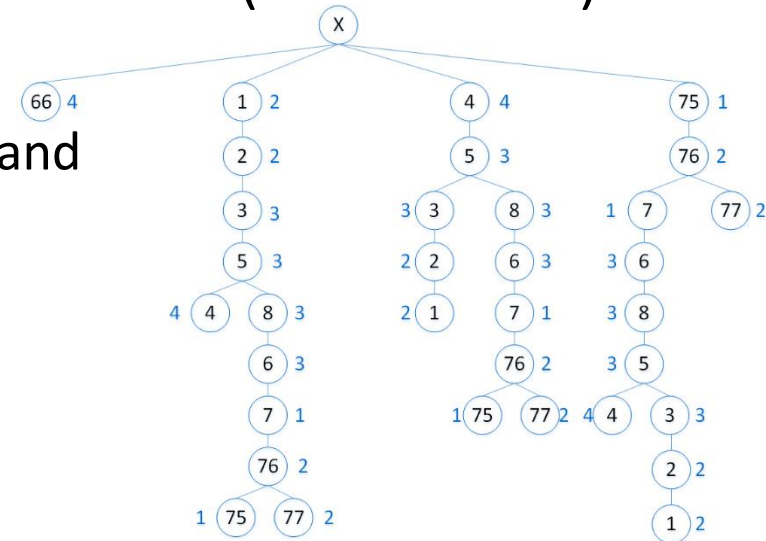
- Solange die Schlange nicht leer ist

➡ • Wenn der aktuelle Status ein Zielstatus ist

- Erfolg

➡ • Sonst, füge alle möglichen Wege zur Queue inkl. deren Kosten hinzu

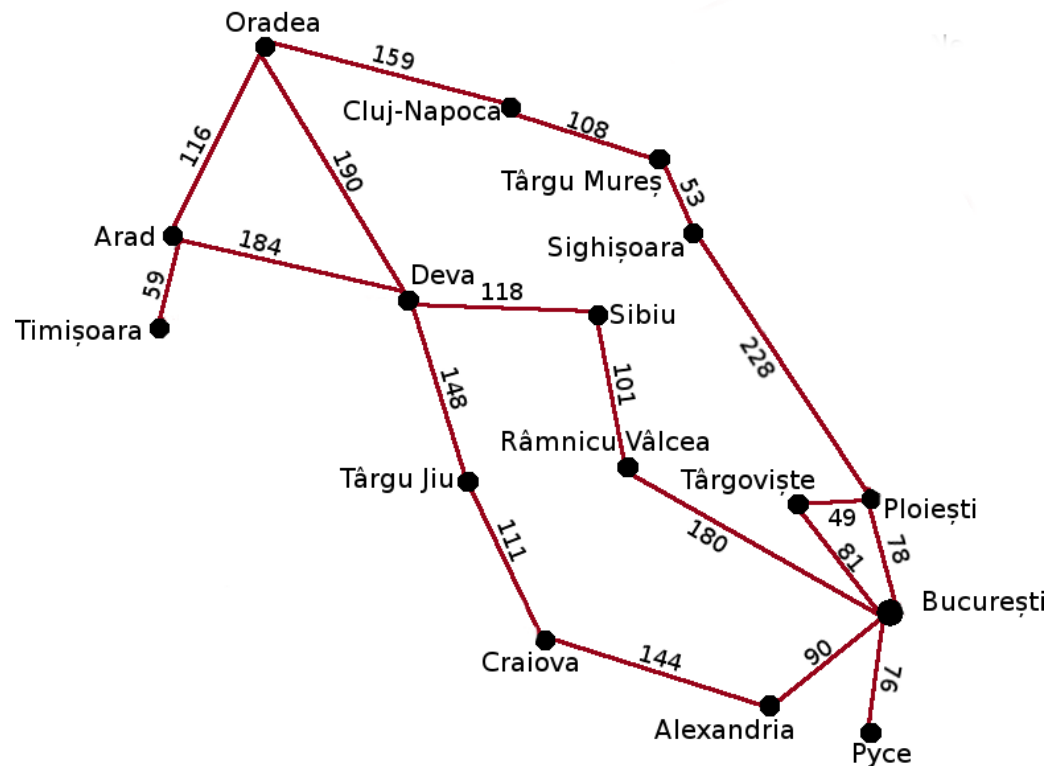
➡ • Sortiere die Priority Queue



Priority Queue:

(X), 0	(X, 66), 4	(X, 1), 2	(X, 4), 4	(X, 75), 1
(X, 75, 76), 3		(X, 1, 2), 4		
(X, 75, 76, 7), 4	(X, 75, 76, 77), 5	(X, 4, 5), 7	(X, 1, 2, 3), 7	

Wende die Uniform Cost Search an, um einen Weg von Oradea nach Bucuresti zu finden



License

You may use and adapt this work for your own non-commercial use.

You may ask the project partners to provide a Moodle course for your own students where you get teacher rights.

You are **not allowed** to make this work publicly available.

When using/adapting the material, you must give appropriate credit to EDLRIS but not in a way that suggests any endorsement. Logos must not be removed.

You may not use the material for any commercial purposes.

If not otherwise indicated in the picture, the images are public domain content from <https://pixabay.com/> or own images and derivative works.