

# Von Labyrinthen zu Algorithmen

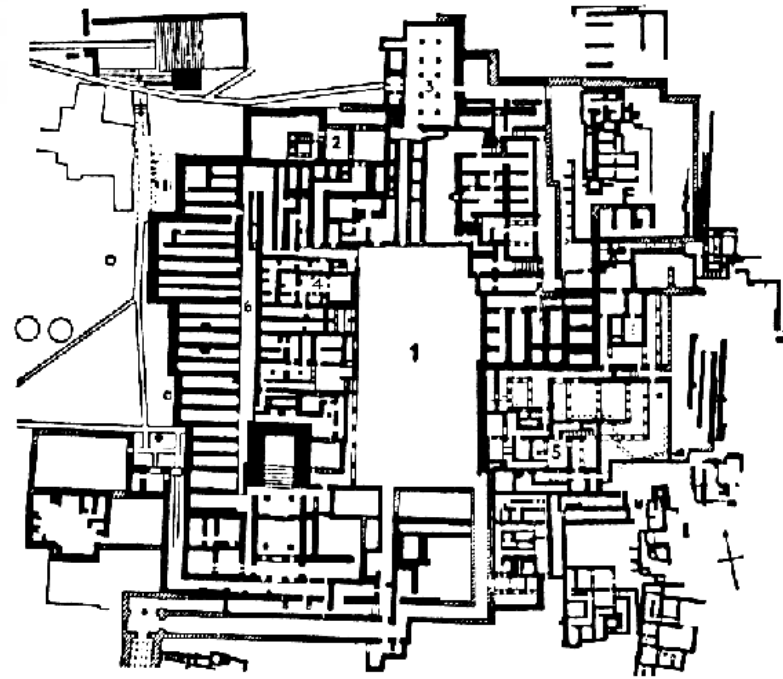
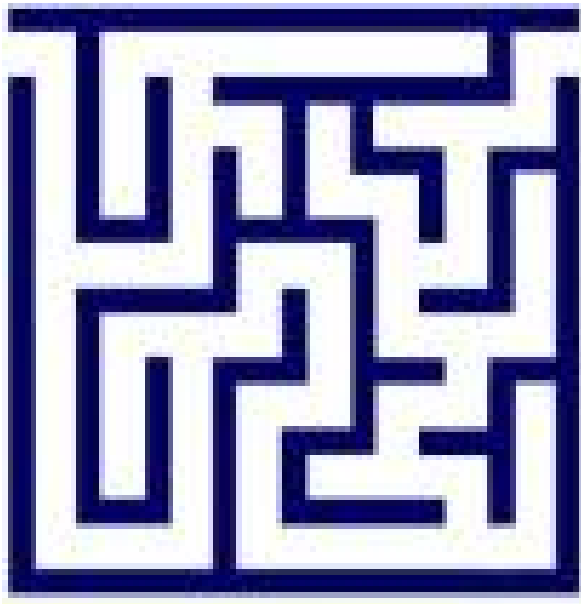
Gerald Futschek

# Wie kommt man aus einem Labyrinth heraus?

- **Labyrinth** (griechisch: Haus der Doppelaxt, wahrscheinlich Knossos auf Kreta)



Labrys

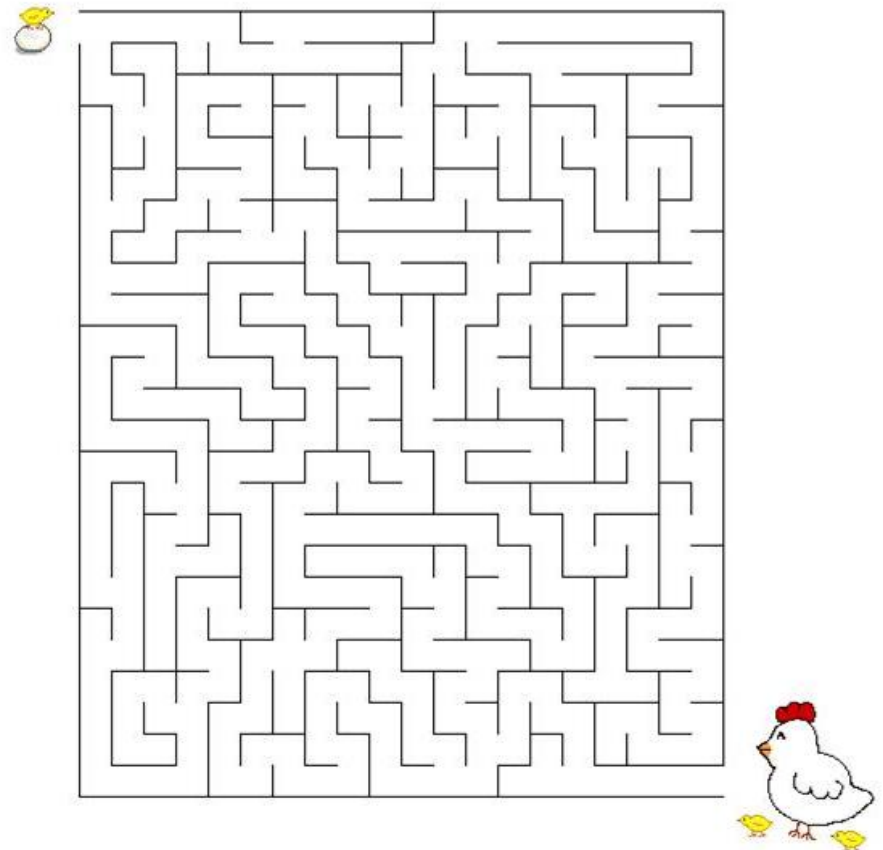


Grundriss des Palastes von Knossos

# Fragestellungen zu Labyrinthen

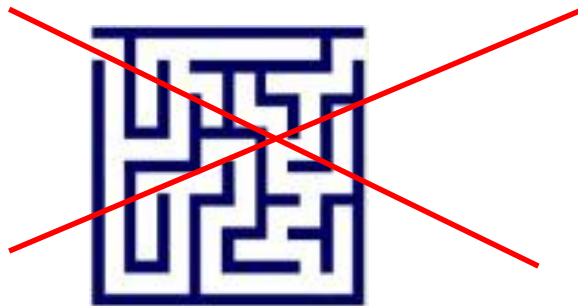
- Finde einen Weg durch das Labyrinth
- Finde einen Weg hinaus
- Finde einen Weg zu einem bestimmten Punkt im Labyrinth
- Gibt es vielleicht mehrere Wege?
- Welcher Weg ist der kürzeste?
- Wie findet man solche Wege?

Aide le poussin à retrouver sa maman.

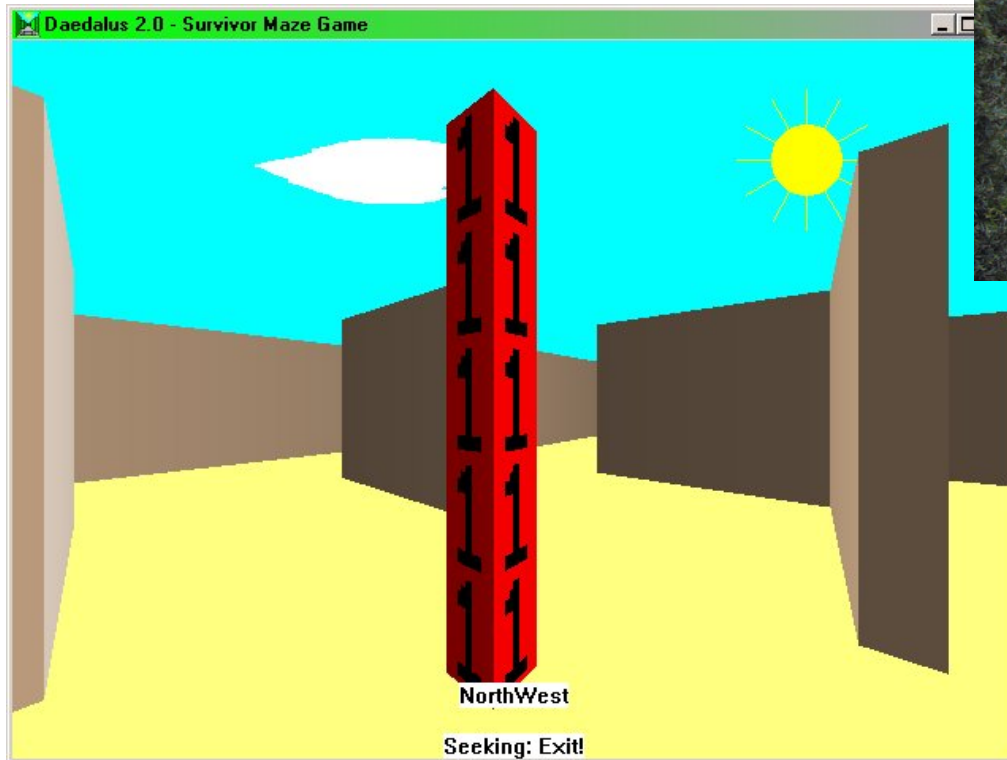


# Präzisierung der Aufgabenstellung

- Gesucht ist ein Weg von Position A nach B
- Form, Größe und Struktur des Labyrinths ist zunächst nicht bekannt



# Sichtweise in einem Labyrinth







Algorithmen

prolog der Informatik

# Suche nach einem Weg

- Ohne zunächst alle Details des Labyrinths zu kennen,
- soll ein bestimmtes Ziel gefunden werden.
- Gesucht ist ein Verfahren, das für alle denkbaren Layrinthe das Ziel findet



# Suche nach einem Weg

- An jeder Kreuzung: Welchen Gang soll man gehen?
  - einen zufälligen?
  - einen bestimmten Gang?
  - alle Gänge der Reihe nach?  
(wie geht das genau?)

Strategie gesucht!

# Eine Idee für eine Strategie

- „Gehe immer der linken Wand entlang!“
- Diese Strategie funktioniert in der Ebene jedenfalls sicher von einem Eingang zu einem Ausgang (Spiegelkabinett)



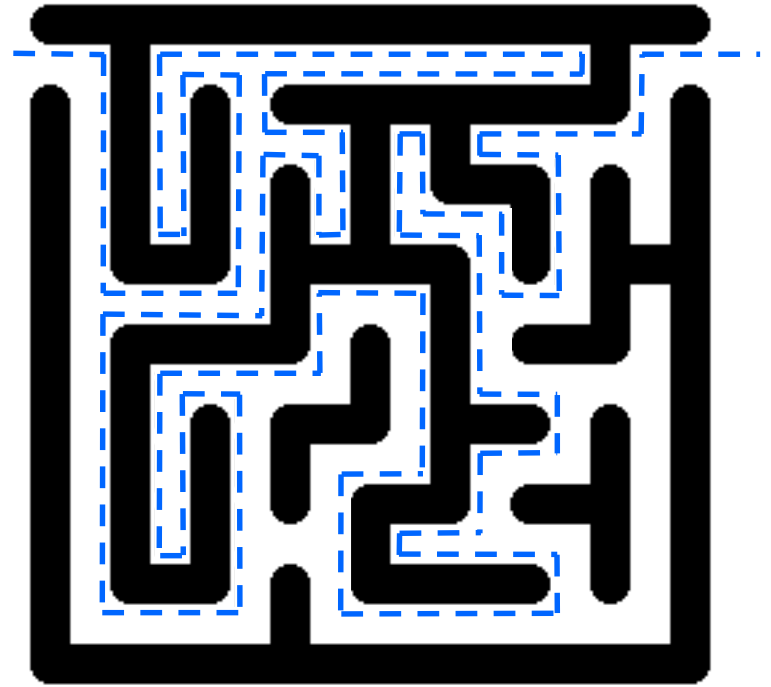
# Eine Lösungsstrategie

- Immer die linke Wand entlang gehen

## Gesucht:

Nicht die Lösung für ein bestimmtes Labyrinth, sondern ein Verfahren, wie man zur Lösung bei beliebigen Labyrinthen kommt!

(ein Algorithmus löst ein allgemeines Problem)



# Abstraktion

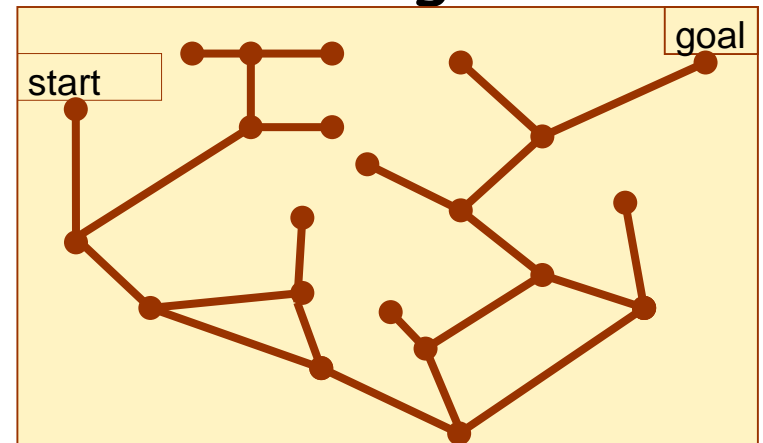
- Labyrinth besteht aus
  - Kreuzungen und
  - Gängen zwischen Kreuzungen
- Länge und Form der Gänge nicht wichtig

Modellierung als **Graph**:

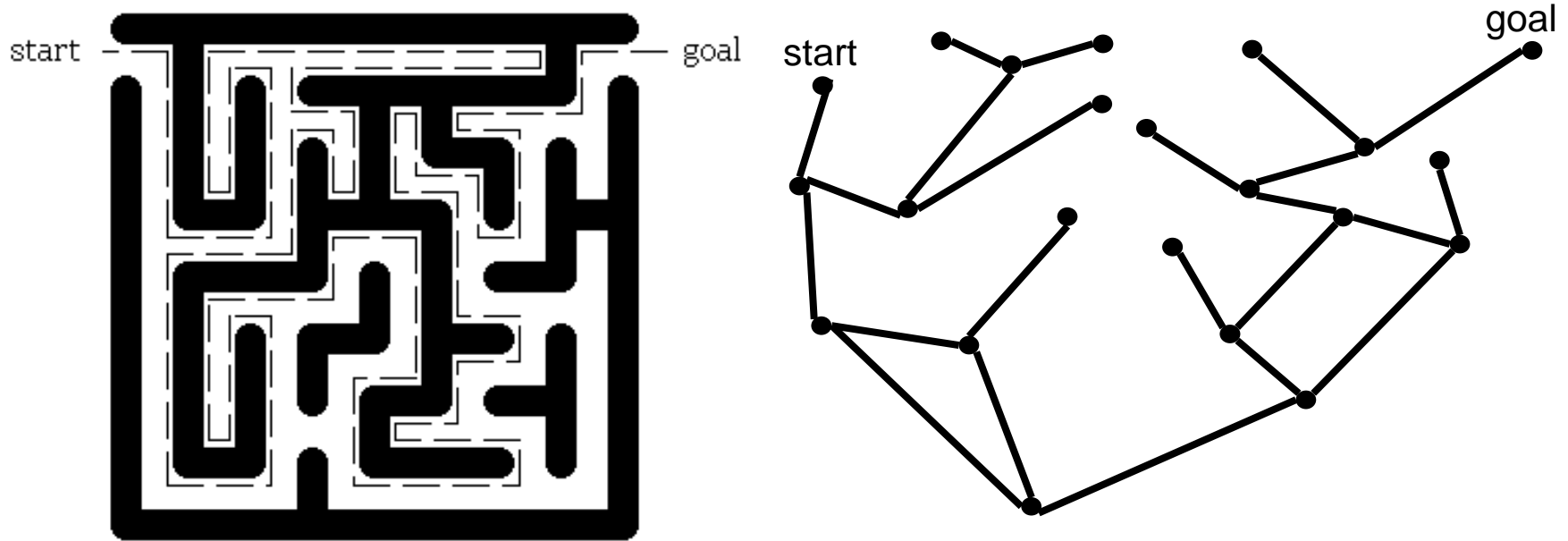
Knoten und Kanten

Kreuzungen werden zu Knoten

Gänge werden zu Kanten



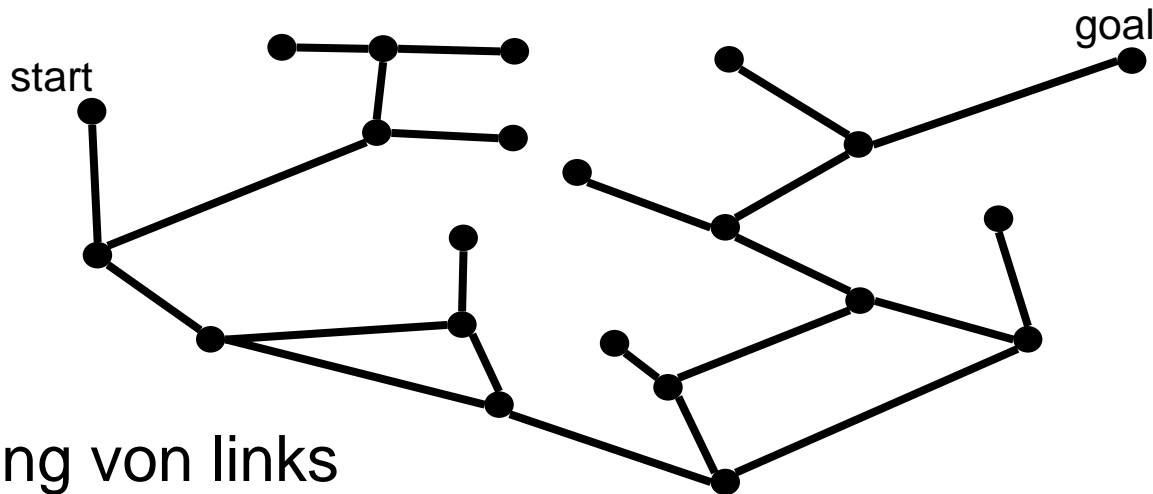
# Labyrinth - Graph



Was entspricht der linken Wand in einem Graphen?

# Der Graph und der 1. Gang von links

- In jedem Knoten müssen wir den 1. Gang von links kennen.
- Welche Kante ist der 1. Gang von links?



Der 1. Gang von links hängt von dem Gang ab, der zur Kreuzung führt!

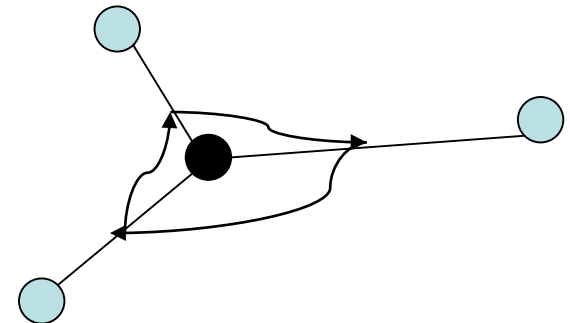
# Modellierung der Reihenfolge der Gänge

- Ein Graph beschreibt nur Knoten und welche Knoten durch Kanten verbunden sind.
- In einem Graph gibt es keine Reihenfolge der Kanten

Für jeden Knoten müssen wir eine Reihenfolge der Kanten zusätzlich modellieren.

Da alle Kanten „gleichberechtigt“ sind, definieren wir gleich eine **zyklische Ordnung**:

Zu jeder Kante  $e$  eines Knoten gibt es eine Nachfolgerkante  $\text{succ}(e)$



$\text{succ}(e)$  ist der  
1. Gang von links!

# Modellierung als Embedded Graph

- Ein **Embedded Graph** (in die Ebene eingebettet) hat die Kanten jedes Knoten zyklisch geordnet
- dh. man hat, wenn man über eine Kante  $e$  zu einem Knoten kommt, mit  $\text{succ}(e)$  den 1. Gang von links
- Mit der zyklischen Ordnung hat man auch eine Reihenfolge, um alle Gänge einer Kreuzung systematisch zu durchlaufen!



# Grundoperationen für Labyrinth

Die folgenden Grundoperationen dürfen in einem Labyrinth-Algorithmus verwendet werden:

Man kommt stets von einem Gang an eine Kreuzung:

- Abfrage: **Anzahl weiterer Gänge bei dieser Kreuzung?** (Null bedeutet Sackgasse)
- Aktion: **Wähle den i-ten Gang von links und gehe in diesem Gang bis zur nächsten Kreuzung**
- Aktion: **Drehe dich um und gehe den Gang, den du gekommen bist, bis zur letzten Kreuzung zurück**
- Abfrage: **Ziel erreicht? Ja/Nein**

Kann man mit diesen Grundoperationen einen Weg von A nach B im Labyrinth finden? Wie?

# Linke Wand entlang

- Wie formuliert man den Algorithmus mit den Grundoperationen?
- Es gibt ja in der Abstraktion (Graph) keine Wände, sondern nur mehr Kanten!

**Algorithmus mit den 4 Grundoperationen:**

**solange** Ziel nicht erreicht

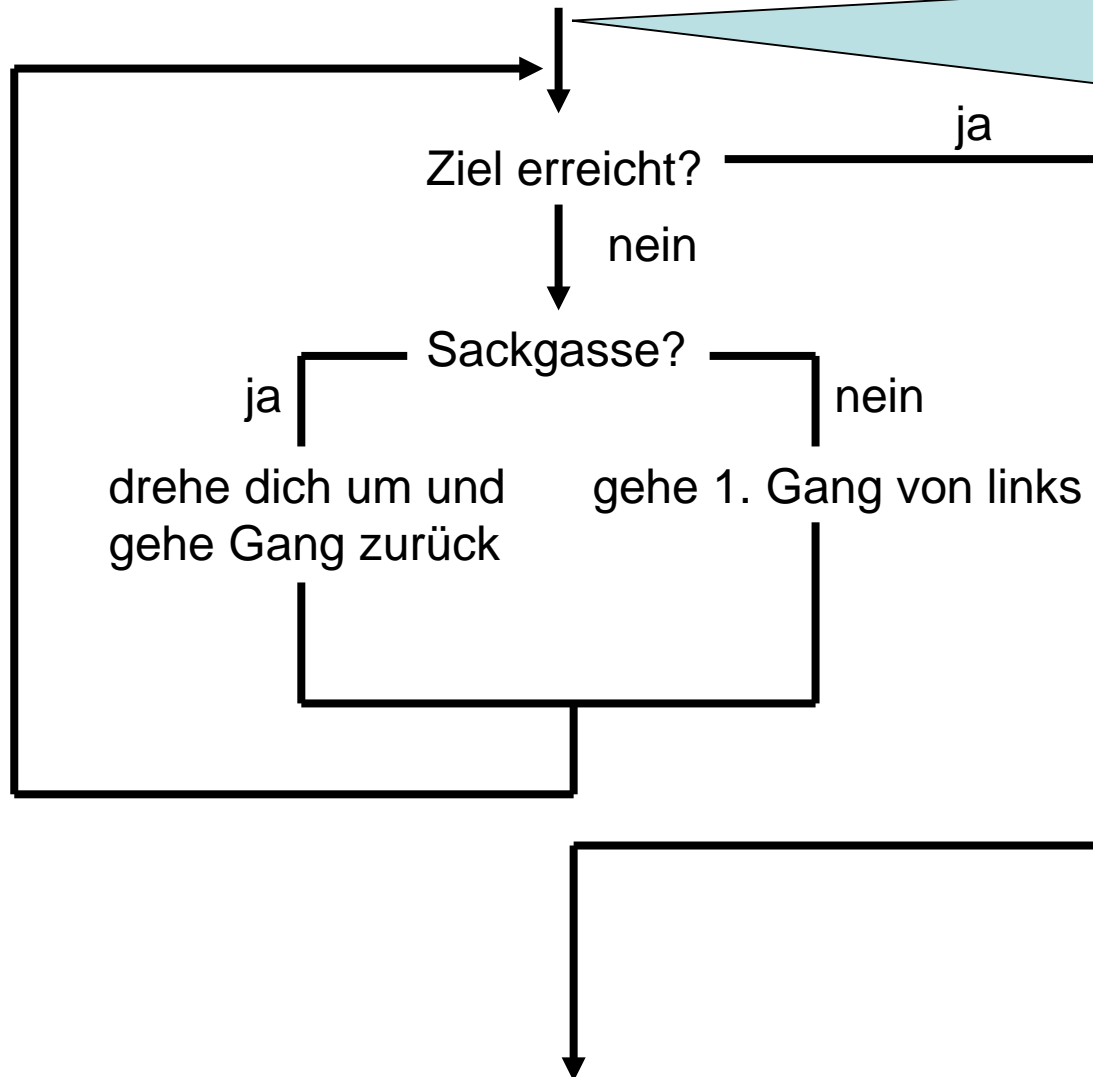
**falls** Sackgasse

    drehe dich um und gehe Gang zurück

**sonst**

    gehe 1. Gang von links

# Flussdiagramm



**Welche  
Anfangsbedin-  
gungen müssen  
erfüllt sein, damit  
dieser  
Algorithmus  
terminiert?**