

Labyrinth

Erzeugung und Wegsuche

Tanja Weber, Matthias Jauernig

Fachbereich Informatik, Mathematik und Naturwissenschaften

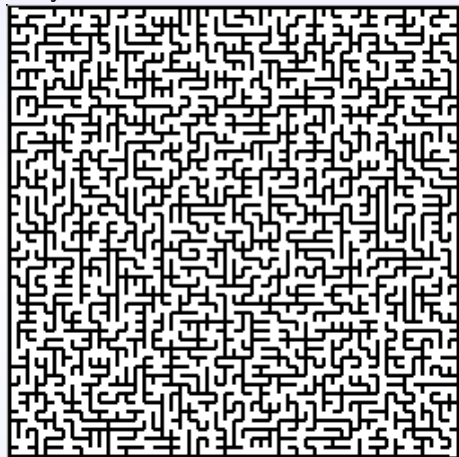
19.11.2007

- Zeitnahme in C# über `QueryPerformanceCounter`
- Durchschnittswerte über 50 Labyrinth
- Pro Labyrinth 10 weitere Durchläufe
- keine Benutzerinteraktionen mit dem Rechner während der Analyse
- Tiefensuche (DFS) iterativ unter Verwendung eines Stacks
- Richtungspräzedenzen: Ost -> Süd -> West -> Nord

- Zeitnahme in C# über `QueryPerformanceCounter`
- Durchschnittswerte über 50 Labyrinth
- Pro Labyrinth 10 weitere Durchläufe
- keine Benutzerinteraktionen mit dem Rechner während der Analyse
- Tiefensuche (DFS) iterativ unter Verwendung eines Stacks
- Richtungspräzedenzen: Ost -> Süd -> West -> Nord

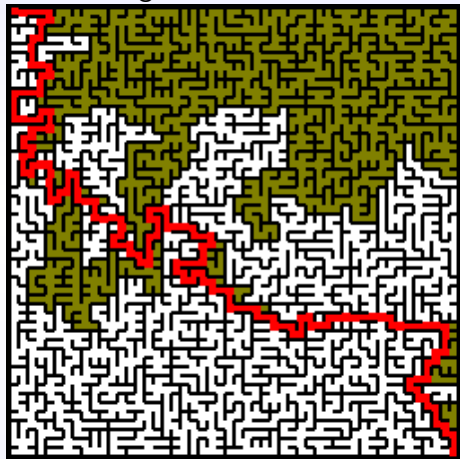
Beispiel-Labyrinth mit Lockerheit 0

- Labyrinth der Größe 50x50:



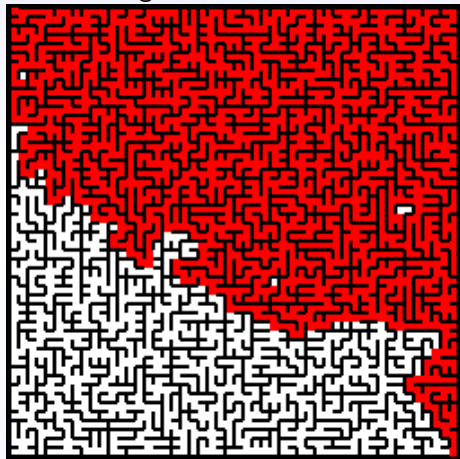
Beispiel-Labyrinth mit Lockerheit 0

- Pfadfindung mit DFS:



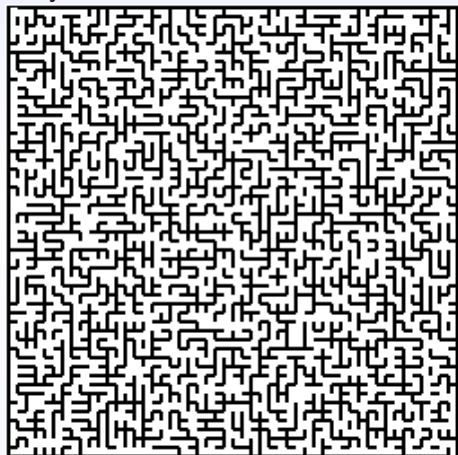
Beispiel-Labyrinth mit Lockerheit 0

- Pfadfindung mit TurnCount:



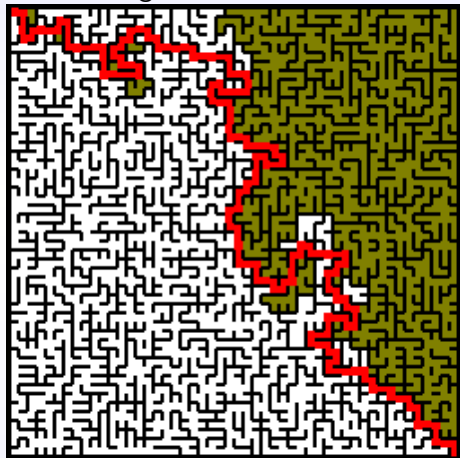
Beispiel-Labyrinth mit Lockerheit 5

- Labyrinth der Größe 50x50:



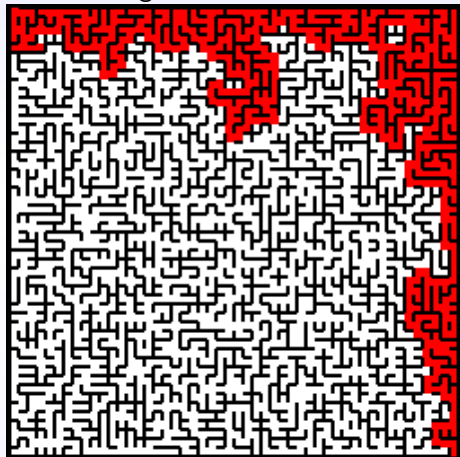
Beispiel-Labyrinth mit Lockerheit 5

- Pfadfindung mit DFS:

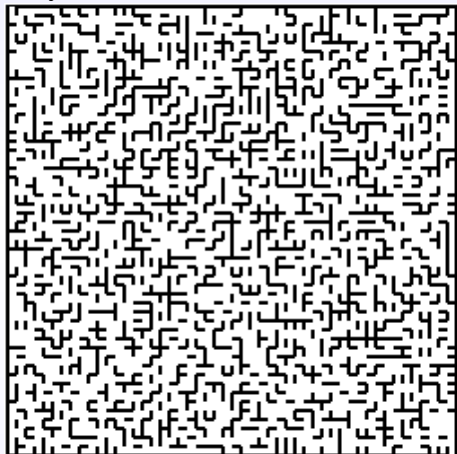


Beispiel-Labyrinth mit Lockerheit 5

- Pfadfindung mit TurnCount:

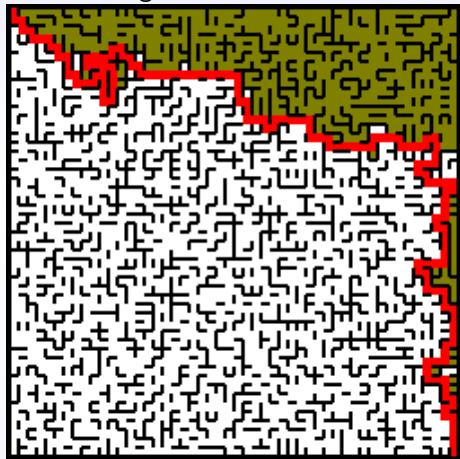


- Labyrinth der Größe 50x50:



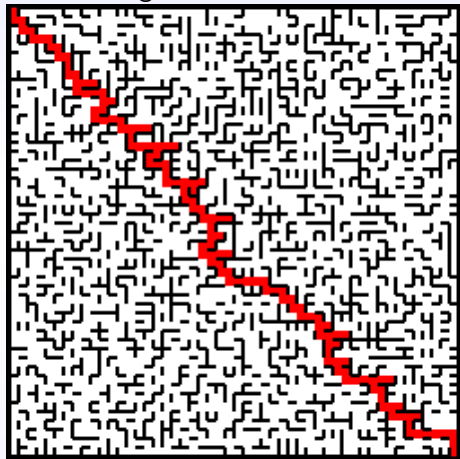
Beispiel-Labyrinth mit Lockerheit 30

- Pfadfindung mit DFS:



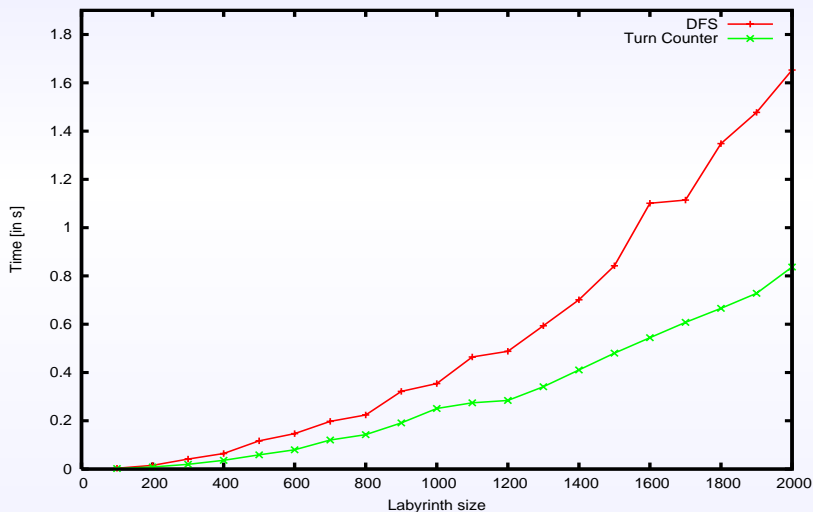
Beispiel-Labyrinth mit Lockerheit 30

- Pfadfindung mit TurnCount:



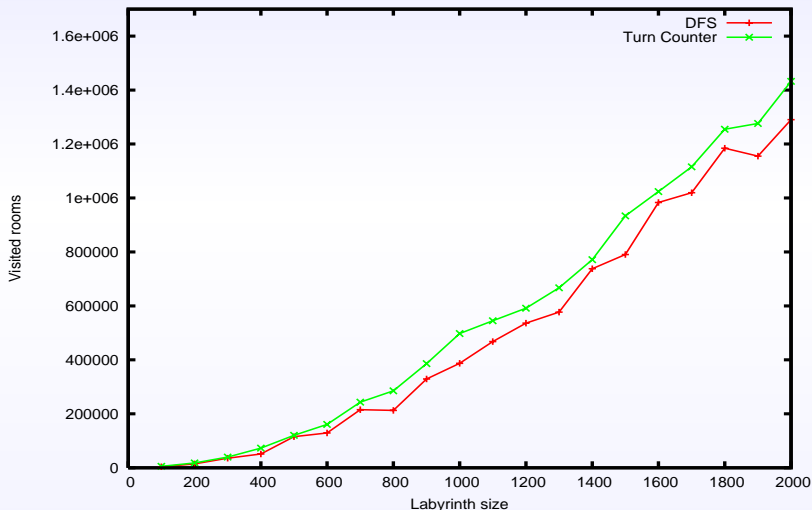
Labyrinthgröße variabel, Lockerheit = 0

- Labyrinthseitenlänge von 100 bis 2000 in 100er Schritten
- Benötigte Zeit:



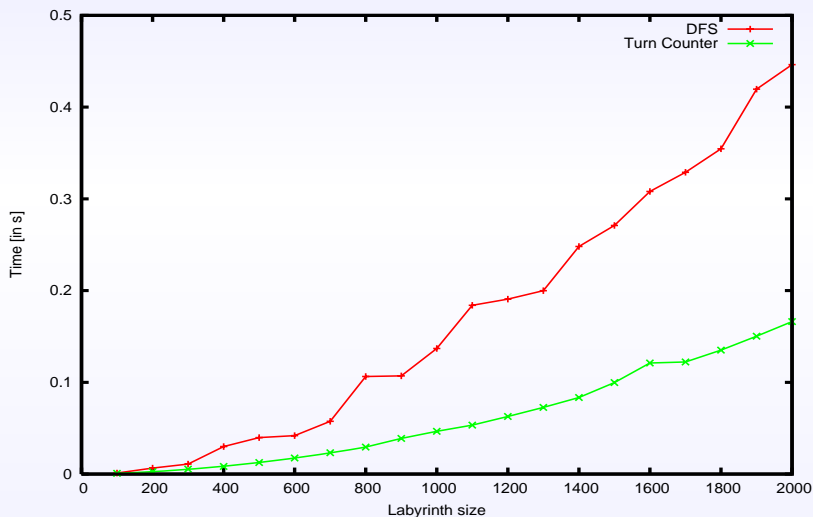
Labyrinthgröße variabel, Lockerheit = 0

- Anzahl besuchter Räume:



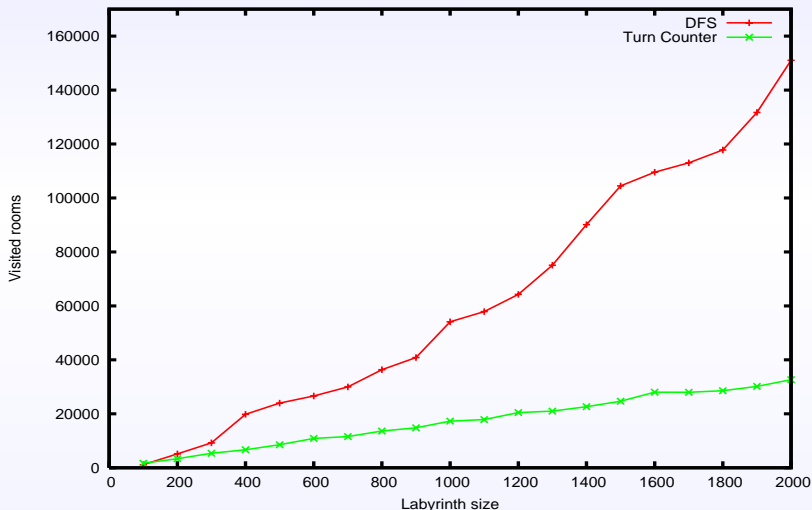
Labyrinthgröße variabel, Lockerheit = 5

- Labyrinthseitenlänge von 100 bis 2000 in 100er Schritten
- Benötigte Zeit:



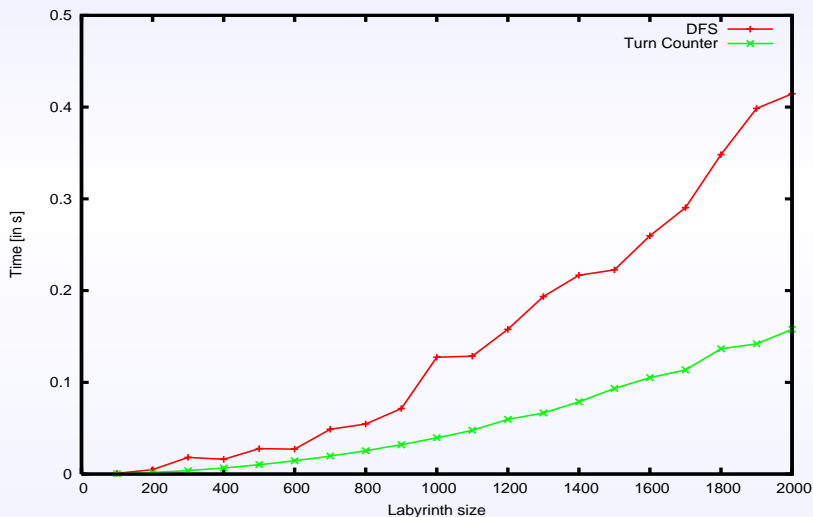
Labyrinthgröße variabel, Lockerheit = 5

- Anzahl besuchter Räume:



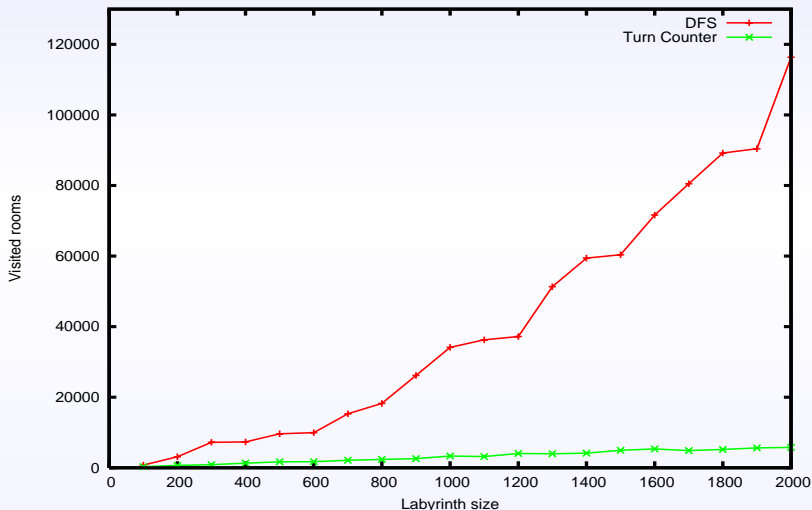
Labyrinthgröße variabel, Lockerheit = 30

- Labyrinthseitenlänge von 100 bis 2000 in 100er Schritten
- Benötigte Zeit:



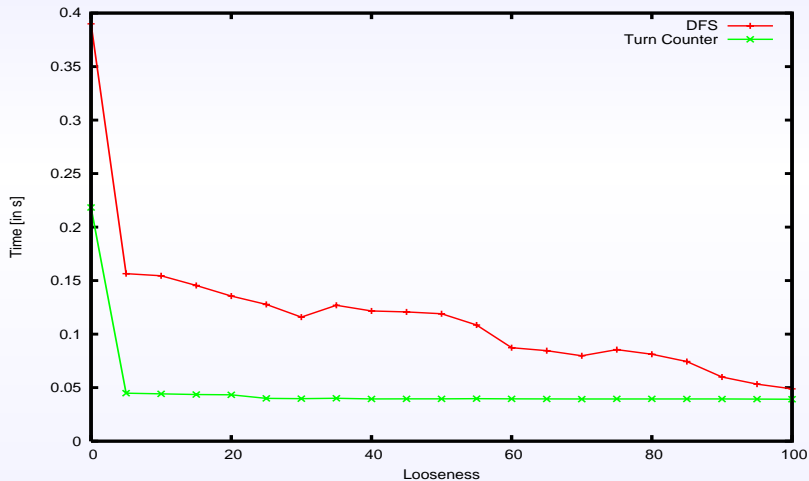
Labyrinthgröße variabel, Lockerheit = 30

- Anzahl besuchter Räume:



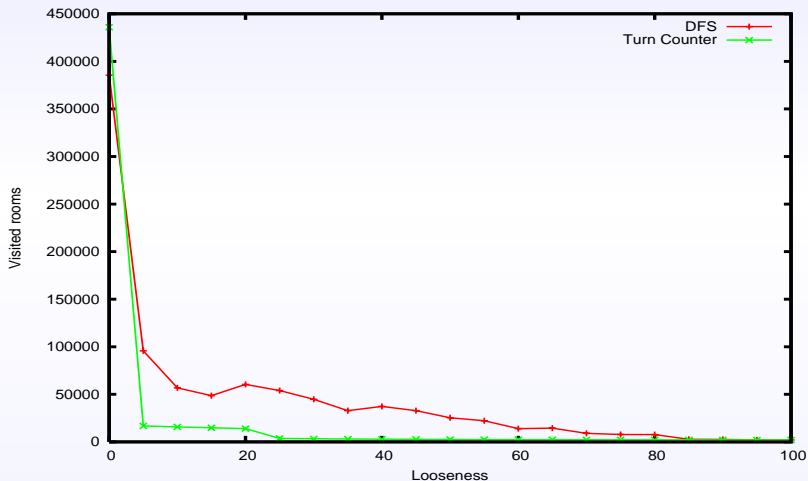
Lockerheit variabel, Labyrinthseitenlänge = 1000

- Lockerheit von 0 bis 100 in 5er Schritten
- Benötigte Zeit:



Lockerheit variabel, Labyrinthseitenlänge = 1000

- Anzahl besuchter Räume:



- Umdrehungszähler nur bei wirklich dichten Labyrinthen in der Anzahl besuchter Räume leicht schlechter als DFS
- Wand-Ablaufen bei Umdrehungszähler bringt nichts/wenig bei dichten Labyrinthen mit nur einem Rundgang
- bei spärlichen Labyrinthen findet der Umdrehungszähler besser den richtigen Weg, auch bedingt durch Geradeauslaufen in Anfangsrichtung
- Vorteil des Umdrehungszählers: keine „Brotkrumen“-Verwaltung
- DFS zu starr durch statische Abarbeitung der Richtungen bei einem Raum
- DFS stärker vom Labyrinthaufbau abhängig, wieder bedingt durch statische Richtungen

- Umdrehungszähler nur bei wirklich dichten Labyrinth in der Anzahl besuchter Räume leicht schlechter als DFS
- Wand-Ablauf bei Umdrehungszähler bringt nichts/wenig bei dichten Labyrinth mit nur einem Rundgang
- bei spärlichen Labyrinth findet der Umdrehungszähler besser den richtigen Weg, auch bedingt durch Geradeauslaufen in Anfangsrichtung
- Vorteil des Umdrehungszählers: keine „Brotkrumen“-Verwaltung
- DFS zu starr durch statische Abarbeitung der Richtungen bei einem Raum
- DFS stärker vom Labyrinthaufbau abhängig, wieder bedingt durch statische Richtungen