

# Pipe Game

Von Christopher Peter

# Inhalt

- ◇ Motivation
- ◇ Thema
- ◇ Aufbau
- ◇ Probleme
- ◇ Lösungen
- ◇ Zusammenfassung
- ◇ Blick in die Zukunft

# Motivation

- ◇ Leicht zu verstehendes Prinzip für das Spiel
  - ◇ Spieler soll schnell verstehen, was er tun soll
- ◇ Nicht zu leicht zu lösendes Spiel
  - ◇ Spieler soll sich nicht langweilen
- ◇ Visuell ansprechend
  - ◇ Wichtiges soll farbig sein

# Motivation

- ◇ Schöne Partikelanimationen
- ◇ Spiele sollen immer einzigartig sein
  - ◇ Pro Spiel neu generiertes Spielfeld
- ◇ Fenstergröße soll dynamisch sein
- ◇ Endscreen

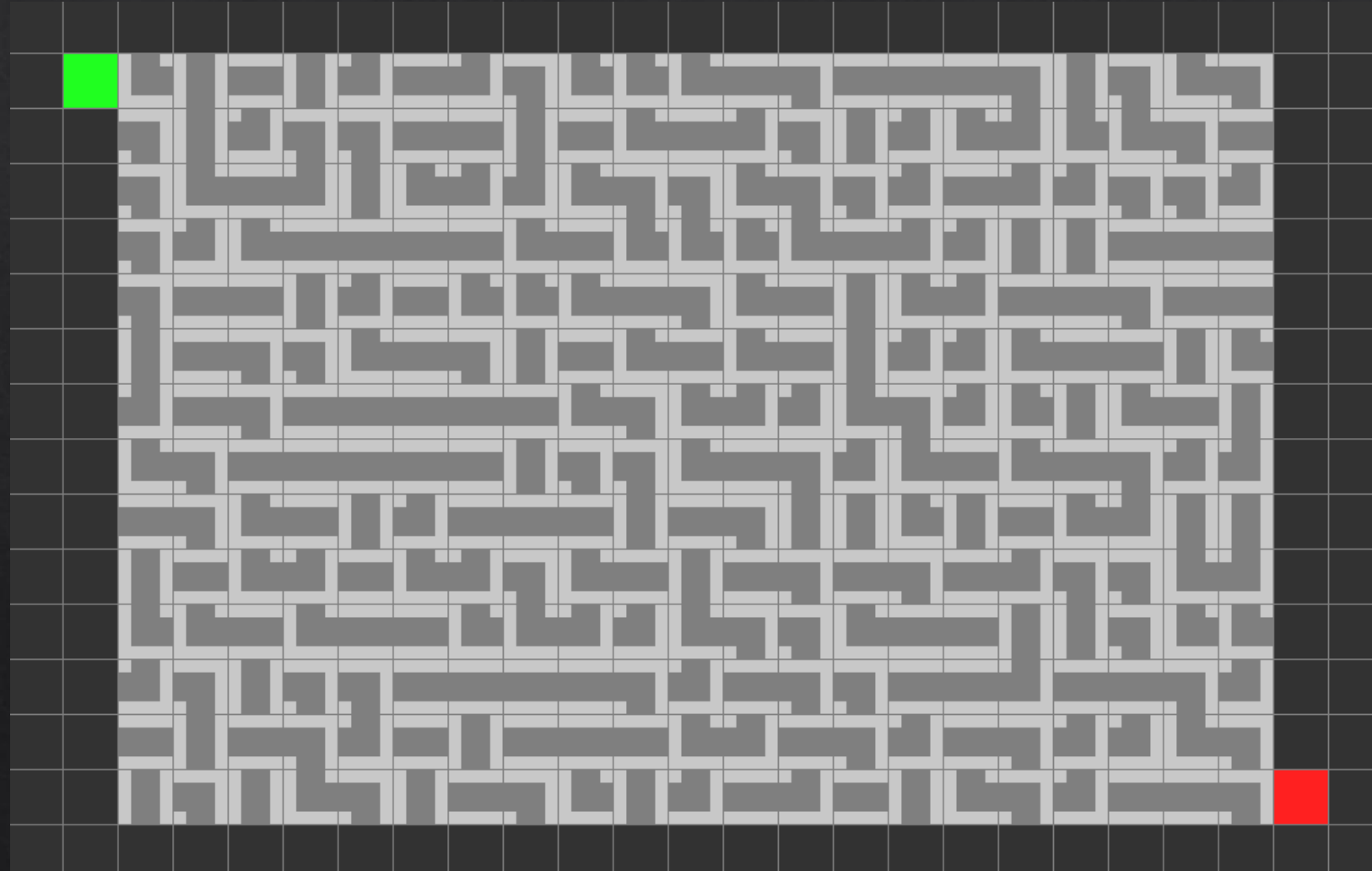
# Thema

- ◆ Idee stammt aus Denkspiel-App
  - ◆ Gleiches Prinzip:
    - ◆ Röhren müssen gedreht werden um farblich markierte Endpunkte zu verbinden
- ◆ Wenn Endpunkte verbunden sind, gewinnt der Spieler



# Aufbau

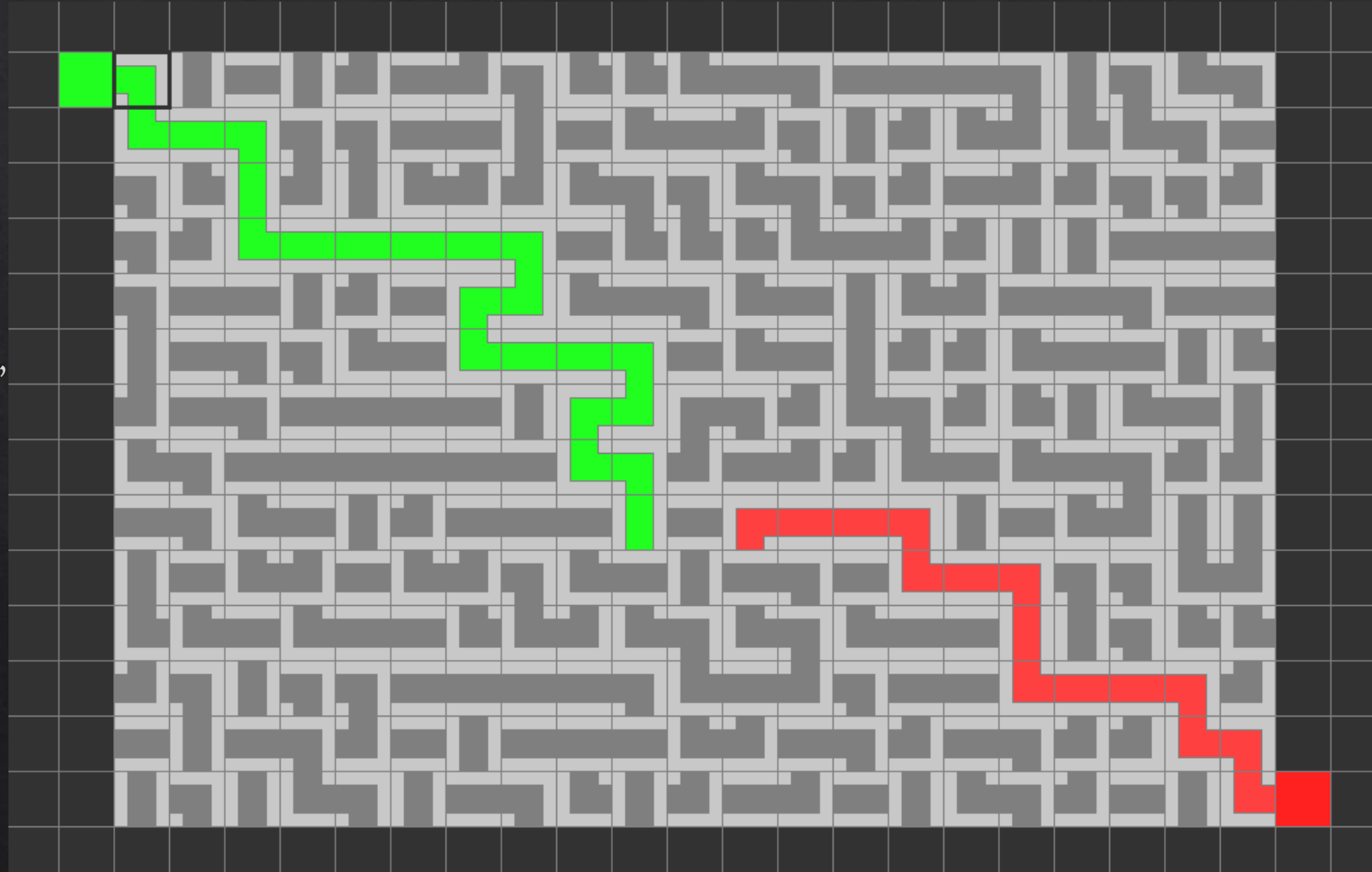
- ◇ Spielfeld besteht aus:
  - ◇ Schwarzem Rahmen
  - ◇ Endpunkten
    - ◇ Grün/Rot
  - ◇ Manipulierbarem Bereich
    - ◇ Hellgrau
  - ◇ Pipes, um Verbindung Herzustellen





# Aufbau

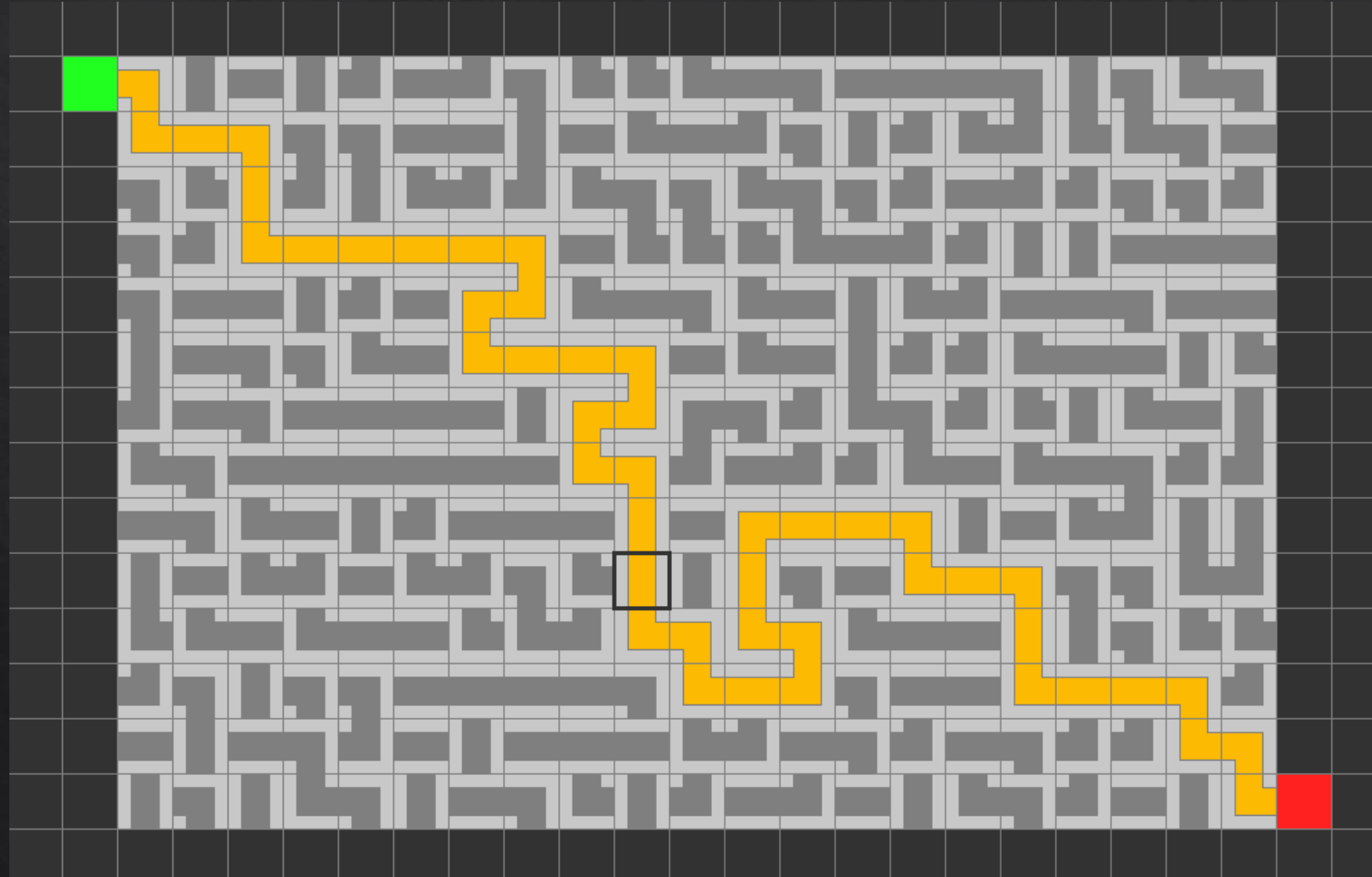
- ◇ Spielverlauf:
  - ◇ Pipes werden gedreht
  - ◇ Farbe fließt, bis Verbindung unterbrochen ist
  - ◇ Sobald Verbindung existiert, kann Farbe weiterfließen





# Aufbau

- ◇ Spielverlauf:
  - ◇ Pipes werden gedreht
  - ◇ Farbe fließt, bis Verbindung unterbrochen ist
  - ◇ Sobald Verbindung existiert, kann Farbe weiterfließen
  - ◇ Ist die Verbindung vollständig, mischen sich die Farben



# Aufbau

- ◆ Spielverlauf:
  - ◆ Pipes werden gedreht
  - ◆ Farbe fließt, bis Verbindung unterbrochen ist
  - ◆ Sobald Verbindung existiert, kann Farbe weiterfließen
  - ◆ Ist die Verbindung vollständig, mischen sich die Farben
  - ◆ Endscreen



# Aufbau

- ◆ Endscreen:
  - ◆ Spieler erzeugt Rakete durch Click an Mausposition
  - ◆ Rakete steigt nach Oben
  - ◆ Rakete explodiert in 1000 Partikel
  - ◆ Farbe im Bereich [R,G,B,C,M,Y]
  - ◆ Partikel verschwinden nach kurzer Zeit



# Aufbau

- ◆ Endscreen:
  - ◆ Zusätzlich: Sterne
    - ◆ Es werden immer 100 Sterne zufällig platziert
    - ◆ Sterne haben zufällige Größe
    - ◆ Sterne „Funkeln“

# Probleme

- ◇ Dynamische Fenstergröße
- ◇ Rechenaufwand minimal halten → mehr FPS
- ◇ Path-Detection-Algorithmus für Hauptspiel
- ◇ Raketen und Partikel implementieren
- ◇ Sterne „Funkeln“ lassen



# Lösungen

- ◆ Dynamische Fenstergröße → Variablen für Positionen, keine Skalare
- ◆ Rechenaufwand minimal halten → mehr FPS → `rect()` anstelle von `ellipse()` für Partikel
- ◆ Path-Detection-Algorithmus für Hauptspiel → Orientierung an AlgoDat (Labyrinth)
- ◆ Raketen und Partikel implementieren → Orientierung an Vorlesungen
- ◆ Sterne „Funkeln“ lassen → Sterne ändern Größe und Opacity



# Zusammenfassung

- ◇ Sinn des Spiels: Drehe die Röhren so, dass eine Verbindung zwischen zwei Endpunkten entsteht.
- ◇ Nach dem Spiel kommt Endscreen
- ◇ Raketen, Partikel, Sterne

# Blick in die Zukunft

- ◆ Mehr visuelle Details im Hauptspiel (tropfen von Röhren, die zu der Verbindung gehören)
- ◆ Drehanimation der Röhren?
- ◆ Audio für Hauptspiel und Raketen
- ◆ Mehr Visuelles für Endscreen (Sternschnuppen)