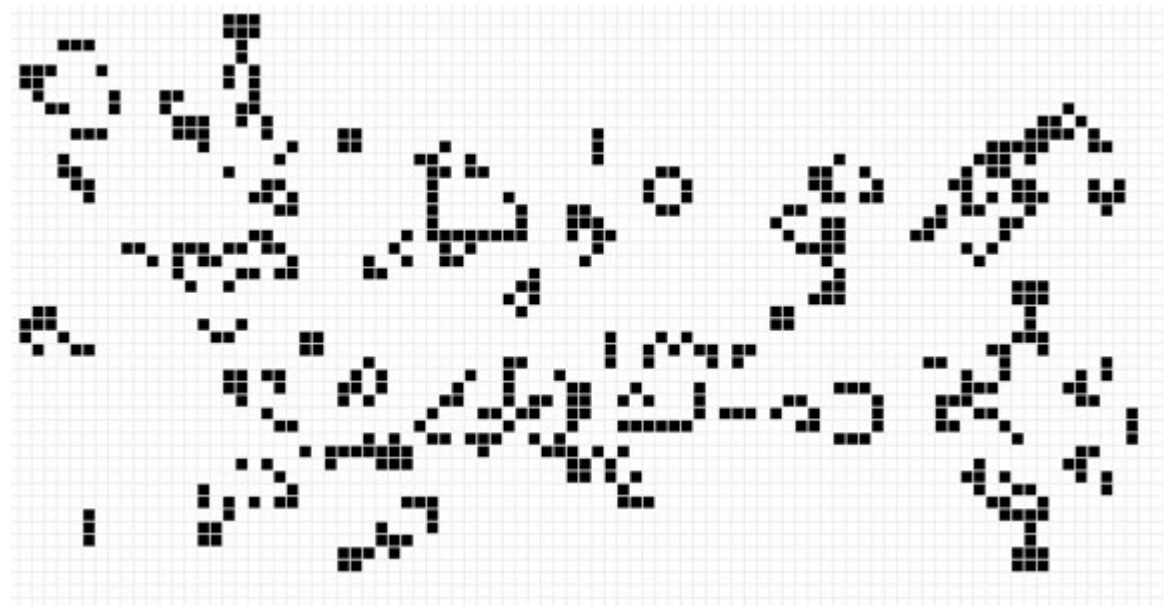




Game of Life





- 1) Motivation**
- 2) Was erwartet mich heute?**
- 3) Test Driven Development**
- 4) Die Aufgabe**
- 5) Organisatorisches**



- **Beispiele aus verschiedenen Berufen:**

- Professioneller **Musiker**:

Übungszeit pro Woche: $5 \times 6h = 30h$ 😎
Performance pro Woche: 2h

- Professioneller **Sportler**:

Übungszeit pro Woche: $6 \times 4h = 24h$ 😎
Performance pro Woche: 1.5h

- Professioneller **Softwareentwickler**:

Übungszeit pro Woche: **0h**
Performance pro Woche: 40h 😓

Was erwartet mich heute?



- **Freiraum zum Experimentieren, Üben, Lernen und Ausprobieren, ...**
- **Neue Ideen und Ansätze sammeln**
- **Besseres Verständnis der eigenen Entwicklungsumgebung**
 - Raum neue Entwicklungsumgebung ausprobieren
- **Reflexion des eigenen Verhaltens**
- **Spaß an der Softwareentwicklung**



Test Driven Development in aller Kürze

TDD

**ALL CODE IS GUILTY
UNTIL PROVEN INNOCENT**

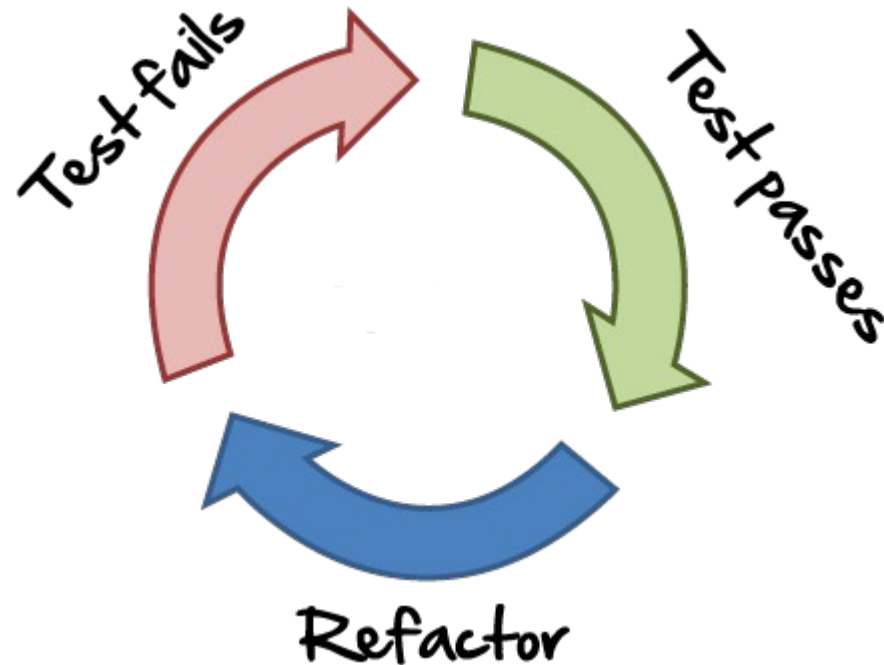
Test Driven Development



Schreibe einen Test („RED“)

Bringe den Test zum laufen („GREEN“)

Optimiere das Design („REFACTOR“)





The Three Rules

1. Test First:

„Schreibe Produktionscode nur für einen **roten** Test“

2. Small Increments (Tests):

„Schreibe gerade soviel Testcode, dass der Test fehlschlägt“
(dazu zählen auch Compilerfehler)

3. Small Increments (Produktionscode):

„Schreibe nur soviel Produktionscode, um einen **roten** Test **grün** werden zu lassen“



- 1) Motivation ✓
- 2) Was erwartet mich heute? ✓
- 3) Test Driven Development ✓
- 4) Die Aufgabe
- 5) Organisatorisches



Spielfeld

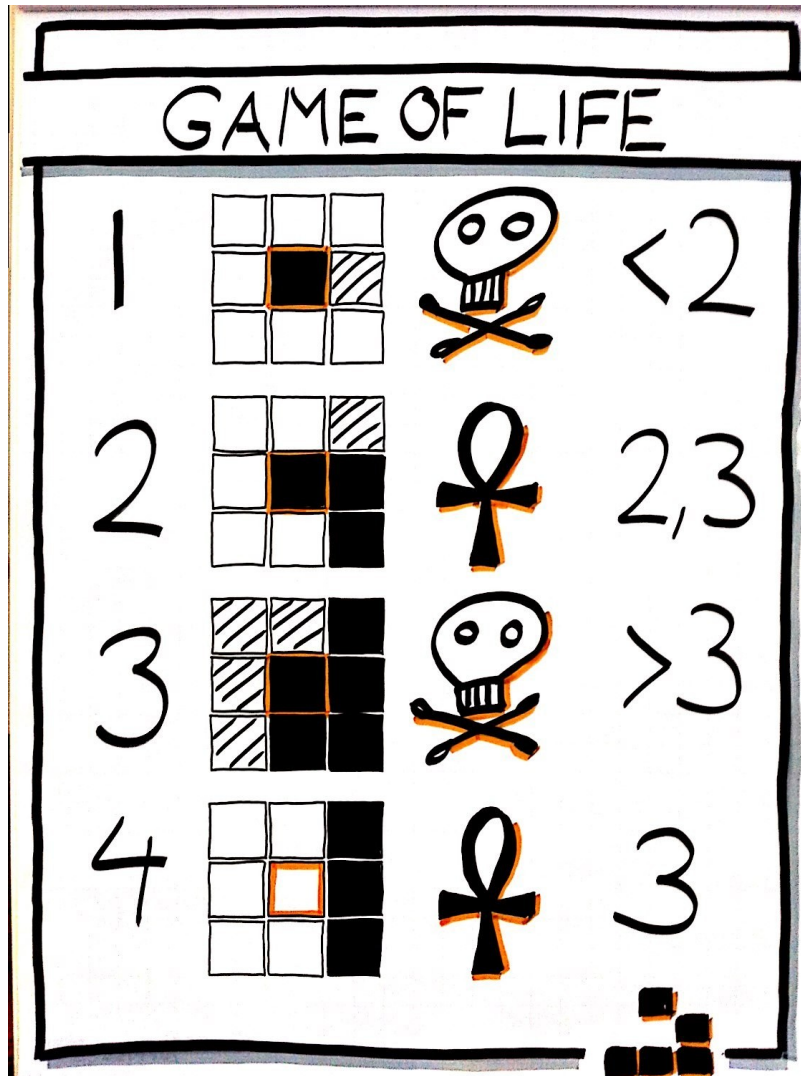
- Spielfeld unterteilt in Zeilen und Spalten → Inhalt sind Zellen
- Zelle hat zwei Zustände → Lebendig oder Tot
- Zufälliger Startzustand zu Beginn



Die Aufgabe: Game Of Life



4 Regeln: Berechnen das Spielfeld der nächsten Runde



Underpopulation:

Weniger als 2 lebende Nachbarn → Lebende Zelle stirbt

Stay alive:

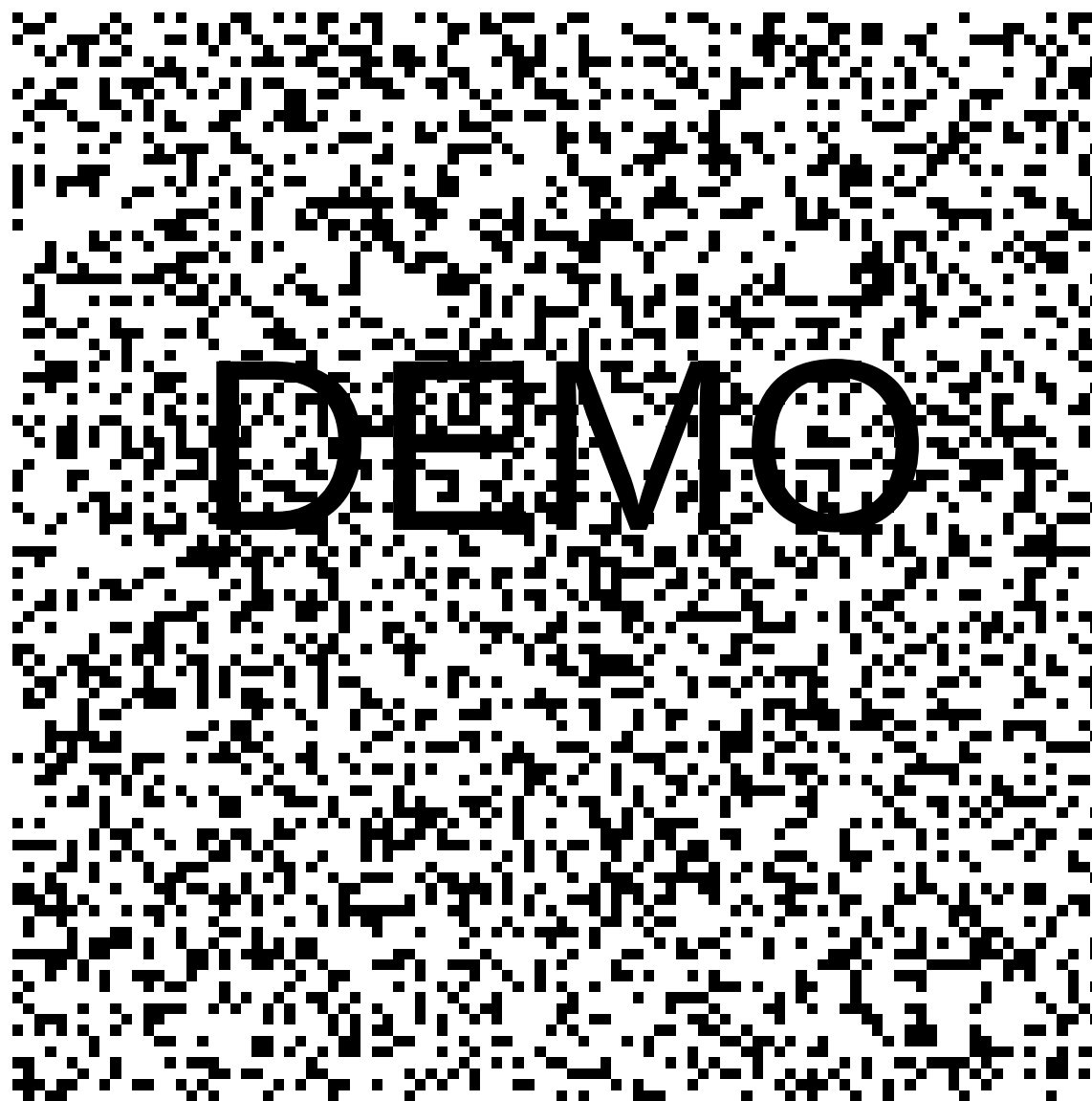
2 oder 3 lebende Nachbarn → Lebende Zelle lebt

Overcrowding:

Mehr als 3 lebende Nachbarn → Lebende Zelle stirbt

Resurrection:

Tote Zelle mit genau 3 lebende Nachbarn
→ Tote Zelle lebt wieder





- 1) Motivation ✓
- 2) Was erwartet mich heute? ✓
- 3) Test Driven Development ✓
- 4) Die Aufgabe ✓
- 5) Organisatorisches



- **Entwicklung mit Pair Programming**
 - 2 Personen entwickeln an einem Rechner
 - 1 Fahrer: Tippt
 - 1 Navigator: schaut voraus, was könnten die nächsten Schritte sein
 - Personen können die Rollen nach X Minuten tauschen
- **4-6 Iterationen zu je 50 Minuten**
 - Am Ende jeder Iteration wird der Code gelöscht
 - Neue Paare nach jeder Iteration
- **Nach der ersten Iteration**
 - Einführung von Constraints (Einschränkungen)
 - Fokus auf bestimmte Aspekte



Fragen?