

Praktikum „SEP: Java-Programmierung“ SS 2019

Aufgabe 4: Qwirkle

Thomas Bunk und Karlheinz Friedberger

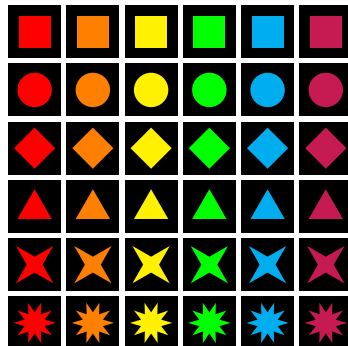
- ▶ Spiel des Jahres 2011
- ▶ Kombinationsspiel für zwei bis vier Spieler
- ▶ Ziel: Durch geschicktes Bilden und Erweitern von Reihen gleicher Farbe oder Form möglichst viele Punkte sammeln
- ▶ Für das Praktikum: Als Mehrspieler-Spiel über Internet (Client-Server Architektur)



Bild: Amazon

Packungsinhalt des Spiels

- ▶ 108 quadratische Spielsteine
 - ▶ die abgebildeten 36 Steine jeweils dreimal
 - ▶ Steine in sechs unterschiedlichen Farben und Symbolen
- ▶ 1 Beutel



- ▶ Spielziel: Sammeln von Punkten durch sinnvolles Auslegen der Steine
- ▶ Spielende: Sobald ein Spieler seinen letzten Stein ausgespielt hat und keine weiteren mehr verfügbar sind
 - ▶ Gewinner ist der Spieler mit den meisten Punkten

- ▶ Benötigt werden ein Stift und Papier zum Aufschreiben der (Zwischen-)Ergebnisse
- ▶ Alle Spieler ziehen verdeckt sechs Steine, sodass diese kein anderer sieht
- ▶ Anschließend ermitteln alle Spieler die Anzahl an Steinen mit gleicher Farbe oder Form.
 - ▶ Der-/Diejenige mit den meisten Übereinstimmungen legt diese in einer Reihe oder Spalte in der Tischmitte aus und beginnt somit das Spiel
 - ▶ Danach wird im Uhrzeigersinn weitergespielt

Der Spielverlauf

Während eines Spielzuges hat der Spieler genau **eine** der beiden folgenden Möglichkeiten:

1. Er legt einen oder mehrere Steine an bereits ausgespielte Steine auf dem Spielfeld an. Anschließend werden die Steine auf der Hand wieder auf 6 (aus dem Beutel) aufgefüllt.
2. Alternativ tauscht der Spieler 1 bis 6 Steine auf der Hand aus, um diese durch neue aus dem Beutel zu ersetzen

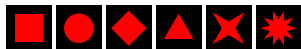


Anlegen von Steinen auf dem Spielfeld (1/2)

- ▶ Neue Steine müssen an bereits gespielte Steine angrenzen und eine Reihe (vertikal oder horizontal) bilden
- ▶ Eine Reihe besteht entweder aus Steinen, die ausschließlich von gleicher Farbe **oder** gleicher von Form sind
 - ▶ In einer Reihe gleicher Form müssen alle Steine unterschiedliche Farben aufweisen:



- ▶ In einer Reihe gleicher Farbe müssen alle Steine unterschiedliche Formen aufweisen:



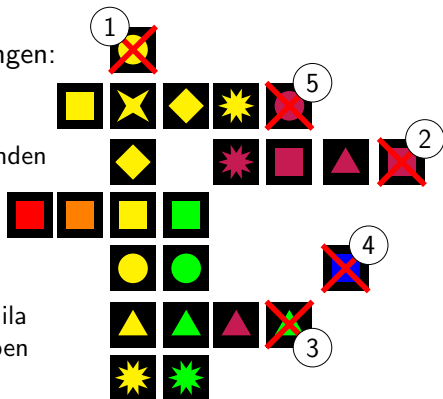
- ▶ Somit maximal 6 Steine pro Reihe möglich!

Anlegen von Steinen auf dem Spielfeld (2/2)

- ▶ Werden Steine an eine bereits existierende Reihe angelegt, so müssen diese in ihrer Eigenschaft zu den bereits ausgespielten Steinen passen.

- ▶ Beispiele für ungültige Platzierungen:

1. Der 7. Stein in der gelben Reihe (zweiter Kreis)
2. Symbol in Reihe bereits vorhanden
3. Analog zu 2.
4. Der Stein hat keinen Kontakt zu den anderen Steinen
5. Der lila Kreis passt zwar zum lila Quadrat, jedoch nicht zur gelben Reihe

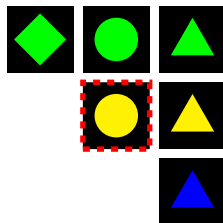


Wertung (1/2)

- ▶ Notieren der Punkte nach jedem einzelnen Spielzug
- ▶ Bilden und ergänzen einer Reihe ergibt für jeden Stein in der Reihe einen Punkt, einschließlich der bereits vorhandenen Steine
- ▶ Ist ein angelegter Stein Teil zweier Reihen, so erhält der Spieler für diesen 2 Punkte

- ▶ Beispiel:

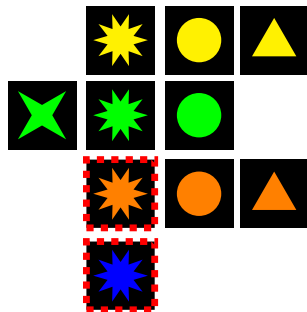
- ▶ Wird der gelbe Kreis angelegt, so erhält der Spieler insgesamt 4 Punkte: 2 für die gelbe Reihe (horizontal), und 2 für die Reihe mit Kreisen (vertikal)



- ▶ Eine Reihe mit 6 Steinen wird *Qwirkle* genannt
- ▶ Das Vervollständigen einer Reihe zu einem Qwirkle ergibt insgesamt 12 Punkte:
 - ▶ 6 Punkte für die 6 einzelnen Steine, sowie
 - ▶ 6 Bonuspunkte für das Erreichen eines Qwirkles

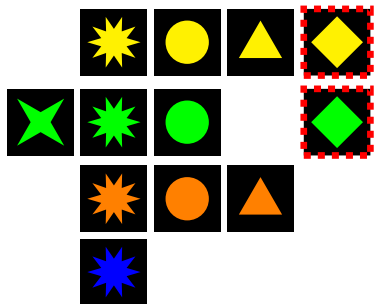
Beispiel zur Wertung (1/3)

- ▶ Ein Spieler legt die rot umrandeten Steine an.
- ▶ Dafür gibt es 7 Punkte:
 - ▶ 4 Punkte für die vertikale Reihe mit Sternen
 - ▶ 3 Punkte für die orange Reihe



Beispiel zur Wertung (2/3)

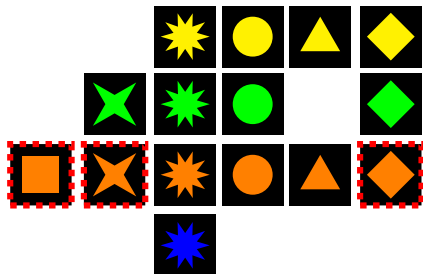
- ▶ Die in diesem Zug markierten Steine geben 6 Punkte:
 - ▶ 4 Punkte für die gelbe Reihe
 - ▶ 2 Punkte für die vertikale Reihe mit Raute-Figuren



Beispiel zur Wertung (3/3)

- ▶ Für die in diesem Zug gelegten und markierten Steine gibt es insgesamt 17 Punkte:

- ▶ 3 für die vertikale Reihe mit Rautesymbolen
- ▶ 2 für die vertikale Reihe mit 4-zackigen Sternen
- ▶ 12 für den Qwirkle (d.h. die horizontale Reihe mit orangen Figuren)



- ▶ Gespielt wird, bis alle verfügbaren Steine in dem Beutel aufgebraucht sind
- ▶ Danach können keine neuen Steine mehr nachgezogen werden. Es wird noch solange weitergespielt, bis der erste Spieler seinen letzten Stein gelegt hat
- ▶ Dieser Spieler erhält 6 Bonuspunkte und beendet damit auch das Spiel
- ▶ Der Spieler mit den meisten Punkten gewinnt das Spiel

WICHTIG:

- ▶ Diese Folien bieten nur einen groben Überblick über die Spielregeln.
- ▶ Die offiziellen Regeln sind (u.a.) zu finden unter https://www.spielregeln.de/wp-content/uploads/2012/01/Spielanleitung_Qwirkle.pdf
- ▶ Diese sind in der Implementierung für das Gruppenprojekt verbindlich umzusetzen

Aufgabenstellung für das gesamte Projekt (1/2)

- ▶ Qwirkle GUI, entweder mit Java Swing oder JavaFX
 - ▶ mit Nebenläufigkeit
 - ▶ Model-View-Controller muss umgesetzt werden
- ▶ Spielbar im Einzelspielermodus
 - ▶ Entweder nur mit sich selbst
 - ▶ oder gegen (bis zu drei) Computergegner
- ▶ Zusätzlich spielbar über Netzwerk mit Sockets
 - ▶ Server wird als eigener Prozess gestartet
 - ▶ Übertragen von Informationen entweder als serialisierte Java-Objekte, oder mittels JSON-Protokoll
 - ▶ Client schickt gewünschte Spielzüge an Server
 - ▶ Server prüft Spielzüge auf Validität
 - ▶ Falls valide: Host leitet Spielzug an alle Clients weiter
 - ▶ Falls invalide: Sender erhält Reject-Nachricht

Aufgabenstellung für das gesamte Projekt (2/2)

- ▶ Ausführliches Testen der Implementierung, z.B.
 - ▶ manuell
 - ▶ Unit-Tests
 - ▶ Integration-Tests
 - ▶ Regression-Tests
 - ▶ ...

- ▶ Gefordert wird eine Testabdeckung in Prozent

Verpflichtende Abgaben

- ▶ Diese Woche (bis Montag, **24.06.2019**):
 - ▶ In LRZ-Gitlab Repo für Projekt einarbeiten
 - ▶ Arbeitsplan erstellen:
 - ▶ Plan für Vorgehen bei der Implementierung, mit vorläufiger Aufgabeneinteilung und evtl. Abhängigkeiten zwischen den Aufgaben
 - ▶ Unbedingt auch schon einarbeiten und mit (prototypischer) Implementierung anfangen
- ▶ Bis Montag, **01.07.2019**:
 - ▶ Voll funktionsfähiges Model
 - ▶ Bedienung mittels einer Shell
 - ▶ Anwendung für einen einzelnen Spieler ohne Gegner
 - ▶ Bei Spielende werden die Punkte des Spielers ausgegeben
 - ▶ Intensiv getestet durch Unit-Tests
 - ▶ Geforderte Test-Abdeckung: 15% (mit IDE-internem Tool)

- ▶ Teams wurden neu zusammengestellt
 - ▶ Größe: 6-7 Teilnehmer
 - ▶ Insgesamt 4 Teams
- ▶ Verlauf der Coaching Meetings
 - ▶ Ergebnisse werden präsentiert
 - ▶ Standup Meeting
 - ▶ Fragen und Antworten
- ▶ Standup Meeting
 - ▶ Meetings brauchen Vorbereitung
 - ▶ Es gibt keine Diskussionen und keine Unterbrechungen
 - ▶ Berichte dem Team über drei Dinge
 - ▶ Was hast du seit dem letzten Meeting getan?
 - ▶ Was hast du bis zum nächsten Meeting vor?
 - ▶ Was sind deine aktuellen Probleme?

Organisation der Teamarbeit

- ▶ In Teambesprechungen sind **alle** gefragt
- ▶ Es besteht Anwesenheitspflicht
 - ▶ Unentschuldigtes Fehlen führt zum Nichtbestehen des Projekts
- ▶ Jedes Teammitglied hat im Arbeitsplan klar definierte Verantwortung, die sich aber ändern kann (mit Begründung)
- ▶ Teamarbeit ist erwünscht, z.B.:
 - ▶ Pair-Programming
 - ▶ Code Reviews
 - ▶ manuelles Testen
 - ▶ Tests für Logik des anderen schreiben
 - ▶ ...
 - ▶ (und natürlich gegenseitige Hilfe bei Problemen)
- ▶ Fragen immer so schnell wie möglich klären, sonst geht viel Zeit verloren

Tipps für die Teamarbeit

- ▶ Kommunikation möglichst informell und häufig gestalten, z.B. via WhatsApp, Slack, Twist, ...
- ▶ Von Beginn an gleichen Codestyle verwenden (z.B. Google Java Style Guide)
- ▶ Gleiche Java-Version verwenden
- ▶ CI nutzen - Unit Tests direkt schreiben