

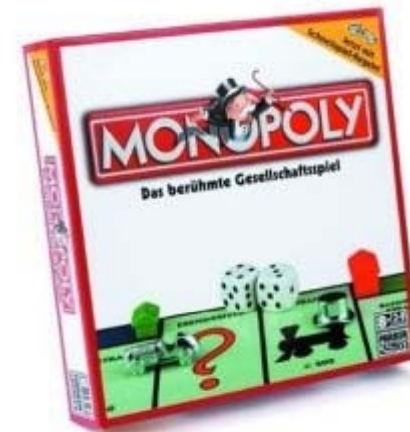
## Gruppenbeispiel – Phase 3





# Aufgabenstellung

- ▶ Erweiterung
- ▶ Refactoring
  - Tutoren-Feedback
- ▶ File I/O
- ▶ Exceptions





## Neue Elemente

- ▶ Companies
  - Electric Company
  - Water Works
- ▶ Railroads
  - Reading Railroad
  - Pennsylvania Railroad
  - B&O Railroad
  - Short Line
- ▶ Jail
  - Spieler darf nicht würfeln
- ▶ GoToJail
  - Schickt Spieler ins Gefängnis



## Im Gefängnis

- ▶ Wann?
  - Go To Jail
  - 3 x Pasch
- ▶ Was passiert?
  - Darf nicht ziehen
  - Darf Häuser kaufen
  - Keine \$200 falls über Go! am Weg
- ▶ Wie lange?
  - Max 3 Runden
- ▶ Wie kommt man raus?
  - Jede Runde:
    - \$50 zahlen
    - Pasch würfeln
      - Kein weiterer Zug
  - Dritte Runde:
    - Würfeln
    - Wenn kein Pasch \$50



## File I/O

- ▶ Spiel kann gespeichert werden
  - Befehl „s“
  - Spiel endet nach Speichern (keine Gewinnabrechnung)
  - Speichert immer in „monopoly.game“
  
- ▶ Spiel kann geladen werden
  - Monopoly mit Kommandozeilenargument „-l“ starten
  
- ▶ Operator Overloading
  - „<<“ und „>>“ von Gamehandler überschreiben
  - File I/O beschränkt sich auf z.B.:
    - `file << game_handler_;`
    - `file >> game_handler_;`



# Neue Fehlermeldungen

- ▶ `ERROR: cannot open saved game  
(monopoly.game)\n`
  - wenn das Laden eines gespeicherten Spiels fehlschlug  
(Kommandozeilenparameter `-l`).
- ▶ `ERROR: cannot save game  
(monopoly.game)\n`
  - wenn das Speichern eines Spiels fehlschlägt.
- ▶ `ERROR: out of memory\n`
  - Wenn kein Arbeitsspeicher mehr zur Verfügung steht.  
**Ausnahme:** in diesem Fall wird nicht der Prompt angezeigt  
sondern das Programm beendet.



## Anhänge zur Spezifikation Phase 2

- ▶ Der Spielstein eines bankrotten Spielers ist **nicht** vom Feld zu entfernen.
- ▶ Sämtliche Beträge, welche ein Spieler durch das Spiel (nicht von anderen Spielern) bekommt werden vom Konto der Bank abgebucht.
- ▶ Sämtliche Beträge, welche ein Spieler durch das Spiel zu zahlen hat (nicht an andere Spieler) werden auf das Konto der Bank gutgeschrieben.
- ▶ Wie die einzelnen Spieler kann auch die Bank im Falle eines Bankrotts einen negativen Kontostand haben.
- ▶ Der Name des Startfeldes ist „Go!“, nicht wie in der Tabelle des Anhangs „GO“.



# Rahmenbedingungen

- ▶ Automatische Tests
- ▶ Abgabegespräche mit Tutor
  - 12.06. - 18.06.
- ▶ Bewertung
  - 14 Punkte Stil
  - 12 Punkte Funktionalität
- ▶ Designdokument
  - Bis 22.5.2008 23:59
- ▶ Abgabe
  - Bis 5.6.2008 23:59
  - gmias.iicm.edu
    - Quellcode
    - Arbeitsaufteilung
    - Makefile
  - gepackt
    - .tar.gz
    - .zip



## 5 Bonuspunkte

- ▶ Als Gruppe zu erreichen
- ▶ Zusatzfunktionalität implementieren
- ▶ Wird **nicht** automatisch getestet
- ▶ bonus.txt in Abgabe
  - Welche Funktionalität wurde implementiert
  - Welche Teilnehmer haben mitgearbeitet
- ▶ Bewertung:
  - Vorhanden / nicht vorhanden
  - Maximal 8 Bonuspunkte insgesamt



# Zusatzfunktionen

- ▶ Wähle 2 aus 5
  - Chance & Community Felder
    - Zufällig gewählter Effekt aus „Kartenstapel“
  - Hypotheken
    - $h \times y \setminus n$
    - Unbebaute Straßen
    - 75% des Kaufpreises
    - 10% Zinsen
  - Häuser verkaufen
    - $d \times y \setminus n$
    - 50% des Kaufpreises zurück
    - Gleichmäßig
  - Versteigerung
    - Nicht gekaufte Straßen werden an alle Spieler versteigert
    - Ausrufpreis: halber Nennpreis



**VIEL ERFOLG!**