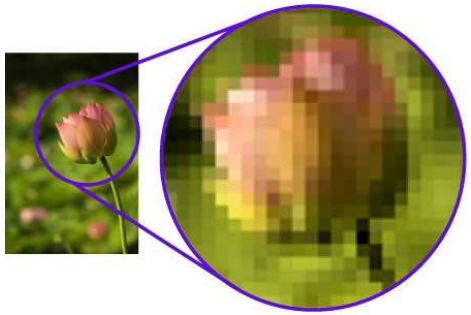
Computergrafik

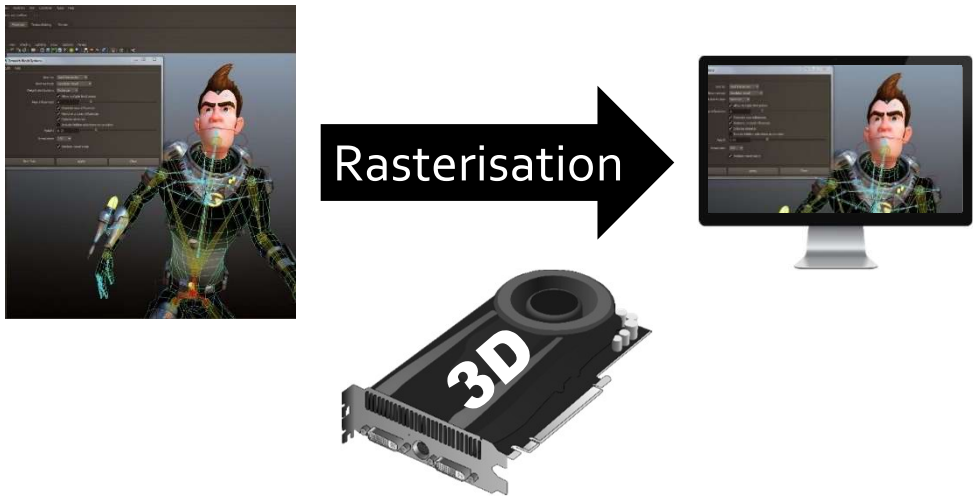
Spiele 2D

Hochschule
Ravensburg-Weingarten
Technik | Wirtschaft | Sozialwesen

First goal – understand concepts

- 2D computer graphics (hardware internals, API)
  - Transformations
  - Textures (Sprites)
  - Blending
  - Particle Systems
  - Rasterization
  - Anti-aliasing
  - Shader intro
- Games
  - Collision detection
  - Physics





Second goal – practicle work

- If **programming skills**
- Big software project
- 2D sprite-based **game**
- Teams (< 5)
- Optional: project w. SE

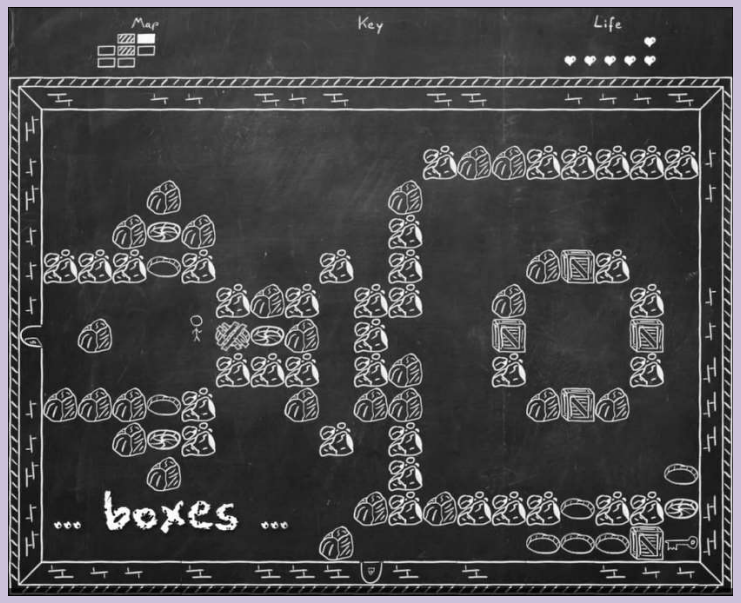
- Graded Exercises
  - Implement theory
  - Mini game
  - Interviews

Necessary Programming Skills

- What is the difference between class and instance?
- What is a static method/variable?
- What is a constructor/method?
- When do I use interfaces?
- How to apply MVC to a given program?
- What are private/public class members and when do I use either of them?
- What is the limiting behavior (big O notation) for insertion/search into a [ordered] list/array/tree?
- What are breakpoints/variable inspection?

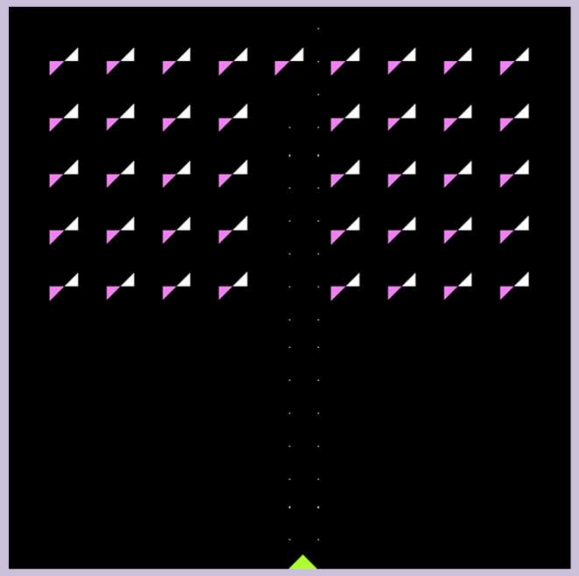
Project – 2D game

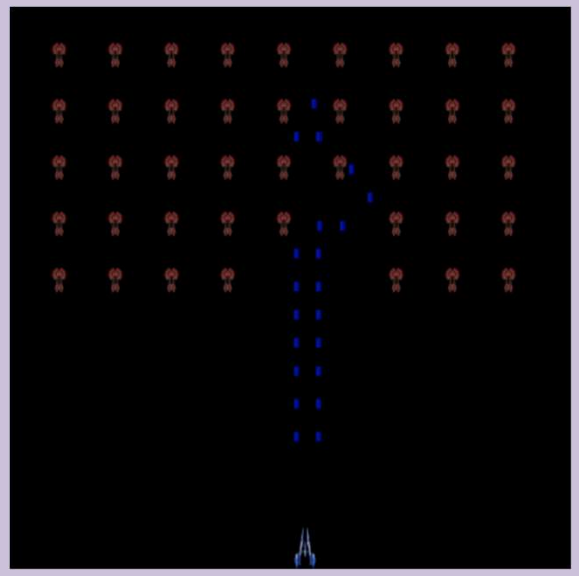
- If existing game → introduce a twist
  - No exact copies allowed!
- **Free** textures/sprites/sounds from web ok

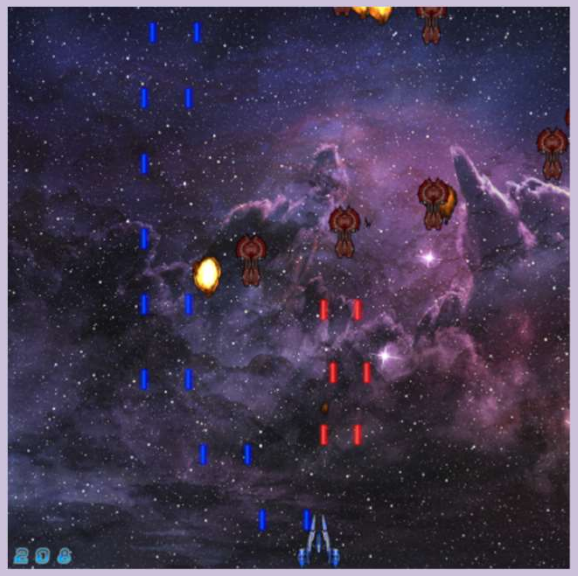


Project – Phases

- Prototype: basic game play, no graphics
- Game play: major game play elements
- Final game: finished graphics and game play







LVA structure

- Work on exercises/project
  - In second part of weekly lecture (help from tutors/me)
  - And at **home**
- Workload 5ects = 150h
  - ~60h spent for lectures + coached working
  - ~90h work at home

Block	Topic	Month 1	Month 2	Month 3	Month 4
1	Lecture				
2+home	Exercises				
	Project	prototype		game play	final game

Todo

- Form teams of 2-4 person(s)
- Create game concept
- Implement a 2D game
  - Team organization
  - Meetings
- Play-testing events
- 1 minute let's play video

- For each exercise
  - Implement steps in readme of exercise
  - Understand code and used theory for interview
  - Ask tutor/myself for interview
  - Receive points

Grading

- Play-testing after each project phase
- Team receives points for each play-testing
- One final grade for team
  - Members distribute grade within team

- Interviews
- We ask questions
  - Used theory
  - Code understanding
- Receive points for each exercise

Programming framework

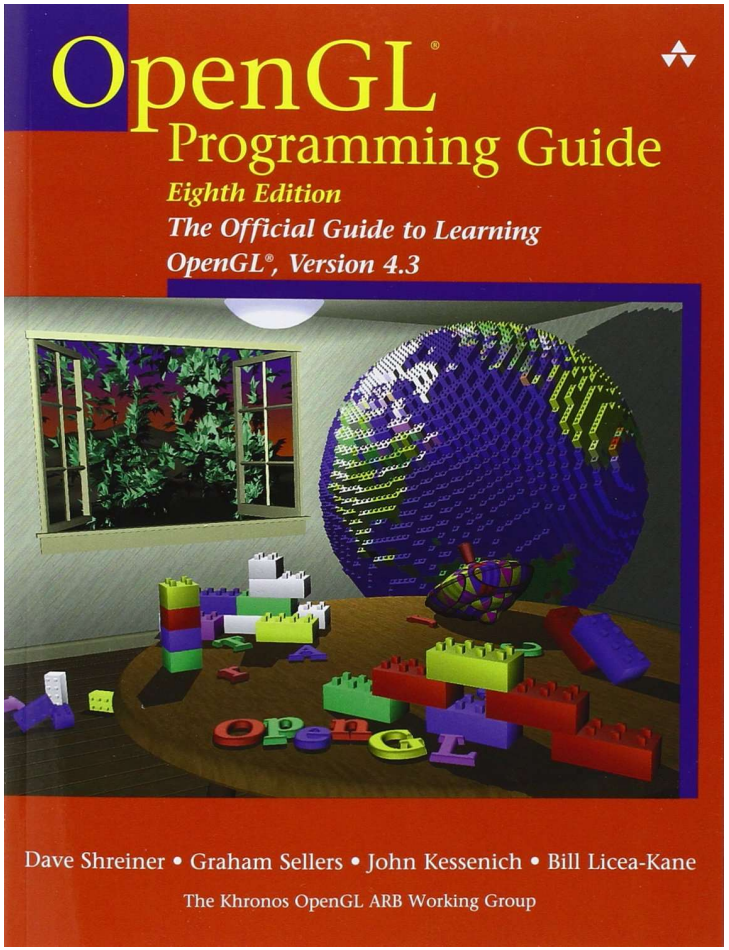
- C#
  - Mix of Java and C++
- MS Visual Studio
  - Linux/MacOS guys can use mono (we cannot give support)
  - Final game should build in Visual Studio
- Graphics: OpenGL graphics API
  - OpenTK which is C# wrapper for OpenGL
- Sound: Naudio
- Zenseless framework (nuget.org/packages/Zenseless)
- Additional libs check with me first

If you want to do a(nother) game

- Talk/mail me!
- Possible with/without team
  - Informatikprojekt (5ects), MD projekte (5, 10, 15 ects)
  - Fachbereichsprojekt Spiele (3 or 5 ects)
  - Bachelor thesis (15 ects)
  - Master (xxx ects)

Books on OpenGL

- Basic knowledge about OpenGL
  - "Red Book"
  - Free: Google: "redbook pdf"
  - Newer version also contain **shader programming**
  - Latest: 8th Edition
  - Tutorials
    - nehe.gamedev.net



Resources

- Slides (pdf) at goo.gl/fVXySr
- "Zenseless" Visual Studio extension(goo.gl/TCAS2K)
  - Source code and examples on Github (goo.gl/924Jlv)
- Exercises
  - "Zenseless" installs project templates

Resources

- Google Calendar (goo.gl/SySLwF)
  - All "profile Game" Relevant Events
- Commented links on games and computer graphics (goo.gl/PUvaAG)
- Moodle
  - Game project final version upload
  - Current state of your grade
  - Forum for questions