Benjamin Mark *March 20 1988*

Last update on December 5, 2014

contact@benjaminmark.dk • +45 3095 0388 • benjaminmark.dk • www.linkedin.com/in/benjamark
Ravnsborggade 7, 1. tv. • 2200 • Copenhagen N • Denmark

Summary

Game Programmer with a passion for procedural generation, a willingness to experiment and an ability to quickly pick up new skills.

As a recent graduate from IT University of Copenhagen with a master's degree in Games Technology, I have gained valuable skills in many aspects of game development. I have completed various group projects which have given me experience working as part of a team, including experience working with designers to realize their vision. My efforts have resulted in two full games as well as a master's thesis that presents a new way to procedurally generate 3D caves.
With a bachelor's degree in computer science from University of Copenhagen, I also have a solid foundation in standard computer science subjects such as compilers, operating systems, graphics, etc. This allows me to approach the task of creating games with a broad base of knowledge to draw on.

Education

IT University of Copenhagen — Copenhagen, Denmark

M.Sc. in Games Technology *2012 – 2014*

A program focusing on the technical aspects of game development including topics such as game engines, AI and procedural generation. This program also allowed me to work with game designers to create two fully fledged games. These and other major projects can be found on my website.
I also completed a number of smaller projects such as implementing various types of AI in Java, implementing various data mining techniques in Java and implementing an erosion algorithm in C++, which erodes a heightmap while taking into account varying layers of rocks.
Master's thesis title: Procedural generation of 3D caves on the GPU.

University of Copenhagen – DIKU — Copenhagen, Denmark

B.Sc. in Computer Science *2008 – 2012*

A traditional computer science program focusing on standard topics such as algorithms, operating systems, compilers and computer architecture. Relevant projects can be found on my website.
Other relevant coursework includes a peer to peer client implemented in python using sockets, a very simple operating system written in C and a compiler for a subset of Cat written in standard ML.

Oure College of Sports & Performing Arts — Oure, Denmark

Studentereksamen (Pre-university secondary education) *2004 – 2007*

Projects

Space Invaders Clone

14 Day Challenge *Nov '14*

A simple clone of Space Invaders with a few new features, that I made in two weeks as a challenge. The game is written in C++ using nothing but the SDL 2 framework. It features the standard Space Invaders mechanics as well as a simple progression system where the player's weapon is upgraded at certain levels of score.

Procedural Generation of 3D Caves on the GPU

Master's thesis project *Mar '14 – Sep '14*

A tool designed to generate 3D caves using the following technique: First the overall layout for a cave is generated with an L-system. This layout is then used as a guide for a tunnel brush which consists of a metaball that has been distorted by noise. It is built using the Unity3D framework and is a collaboration between myself and another student. My main focus was on designing and implementing the generation of the cave layout in C#. I also worked on some of the detailing functionality which is implemented using a mixture of C# and compute shaders written in HLSL.
The paper resulting from this project is currently under review for the Eurographics 2015 conference.

Broken Shadows

Game Development Project *Feb '13 – Jun '13*

A first person exploration game built in UDK using unrealscript. As part of a 5 person team, my main responsibility was to implement the ledge climbing mechanic which, along with the grappling hook, is central to the way the player traverses the environment. Throughout the project I also participated in development of the initial concept as well as the overall game design process.

Simple 2D Game Engine

Game Engines Project *Nov '13 – Dec '13*

A simple, but fully functional, 2D game engine that I built in C++ as part of a Game Engines course. It features simple 2D physics and collision detection, an animation system, a tile based world and an event system. The rendering system is built on top of the SDL 1.2 framework, which handles the actual sprite blitting.

Leap

Game Design Project *Sep '12 – Dec '12*

A 2.5D platformer built in Unity3D and featuring two separate planes. As part of a 5 man team of programmers and designers, I used C# to implement various aspects of gameplay, mostly focusing on enemy behaviour, save functionality and UI as well as some parts of the character controller. I also participated in the concept development and game design process throughout the project.

Simple 3D Renderer

Foundations of Computer Graphics Project *Jan '12 – Apr '12*

Throughout this project I implemented a simple renderer from the ground up. It is based on a framework that was provided for the course, which facilitates drawing pixels to the screen via glut. On top of this framework, I implemented all the functionality required for a simple renderer: Line and triangle rasterization, viewpoint transformations, Gouraud and Phong shading and bezier surfaces. The entire project is written in C++.

It is possible to download and play the games mentioned above from my website: http://benjaminmark.dk

---

Skills

Programming languages:

- **Experienced with:** C#, C++, Java
- **Have used for at least one project:** C, Python, Standard ML, php/SQL

Frameworks:

- **Unity3D** – Used to create one full game and two other projects.
- **UDK** – Used to create one full game.
- **SDL 1.2** – Used to implement rendering for a 2D game engine.

Natural languages:

- **Danish** – Native
- **English** – Full professional proficiency
- **French** – Elementary proficiency

Interests

- **Reading Warhammer 40K novels**
- **Biking**
- **Seeing new places**
- **Game Design**
- **Sailing**