Personal Statement

As a recent graduate of Edinburgh Napier University, with a BSc (Hons) 2:1 in Computer Games Development I am keen to work in the games and computer graphics industries. Having spent 1 year abroad at the Hoge school van Amsterdam, where I undertook group-project work for G-Star (an international retail company), I have gained a good experience of expectations in a professional work environment. I have a passion for games technologies and specifically graphics. As a result, I utilise my spare time creating programs ranging from game mods to tools to help people modify textures and files.

Portfolio: https://matthewjjenkinson.000webhostapp.com/

Blog: https://test-05.weebly.com/

Qualifications

- BSc (Hons) Games Development 2:1
- A-levels: Mathematics (C), Physics (C), Graphic Design (C)
- 13 GCSEs, grade A-B

Skills

Languages & Frameworks: C/C++, C#, DirectX11, Git, GLSL, HLSL, Java, JavaScript, NodeJS, OpenCL, OpenGL, OpenMP, PHP, Vulkan

Software: CodeXL, Microsoft Office, Photoshop, Trello, Unity, Unreal Engine, Visual Studio

Education

Edinburgh Napier University (2014-2018)

Computer Games Development BSc. (Hons)

Modules Studied: Programming Fundamentals, Mathematics, Human Computer Interaction, Software Development, Computer Systems, Computer Graphics, Software Engineering Methods, Database Systems, Systems & Services, Game Technology, Design Patterns for Games, Automated Games Design, Graphics Programming, Advanced Physics, Virtual Reality in Fashion, Computational Intelligence, Information – Society & Security, Advanced Games Engineering, Concurrent & Parallel Systems.

Through my studies I have developed a broad range of technical skills. I have strong programming skills, can think logically, and have a strong work ethic. I am comfortable both working independently and as part of a team. My effective communication skills have enabled the successful development of my project work, for example, in working within a team to create a 3D real-time-strategy game using OpenGL, I was confidently able to take a key role in development discussions and provide my team with progress updates.

1st Year

In this year I thoroughly enjoyed learning about computing theory including how components in computer hardware work together, as well as assembly language, and how programs are compiled. Additionally, my mathematical studies focussed on logarithms, eigen-values, imaginary numbers, vectors and matrices. I subsequently applied this learning in 3D geometric calculations to render scenes for my 2nd year project. I also applied my newly acquired C, Java and F# skills in multiple projects involving object-orientation and polymorphism, and functional programming.

2nd Year

Having seen first-hand the effect and importance of importing height maps in existing games, I decided to focus this year's project on creating my own implementation of this technique to

enhance my understanding of the technology behind the process. This was a 3D environment that could be explored in real-time. For this I used OpenGL and C++ for the CPU portion of the program and GLSL to execute instructions on the GPU. My finished product was a representation of Edinburgh, using a height map that I generated with an online tool. I also used another texture as a map to determine the positions of streetlights (textured meshes with light sources). In total the scene had close to 400 point-lights that used deferred rendering, a free-camera (perspective), a spotlight, and a sun that changed the colour of light it emitted as it got closer to the horizon (as well as turning off once it was below it). I also used post-processing filters, as well as various materials made up of textures, normal maps and material-properties, to determine the final colour of each pixel sent to the screen. I found the challenges of this project both stimulating and rewarding. I now wish to apply the skills and techniques I learnt from this in a professional development environment.

3rd Year

Studying abroad in Amsterdam, I worked in a team for a real-world client – the fashion company G-Star. My team's task was to create a virtual-reality (VR) fitting room to be used with the HTC Vive. Part of my role was to build the locomotion functionality, which I completed successfully using C++ and blueprints in the Unreal Engine. Additionally, I was tasked with modelling and motion capture work to create the prototype, for which I used a range of tools.

- I worked using the Agile development method and was expected to work to an industrial schedule. At the end of each sprint I confidently presented the prototype's progress to G-Star staff, peers and their clients. I was readily able to answer their technical and project-related questions and discuss my planned work. This experience gave me a good understanding of working in real-life industry which I enjoyed thoroughly.
- As part of this work I conducted market research, interviewing G-Star customers in store regarding their likes and dislikes with existing fitting rooms, and their desired features for a VR fitting room. In addition to developing methods of locomotion, I created new ways for users to interact with UX elements (models of clothing, doors, menus) in the VR fitting room. I also undertook pair-programming, to work collaboratively in creating a broad range of features; this technique enabled our group to maintain consistency within our codebase.
- At the end of the project, the strength of my work was recognised by G-Star who requested the use of my code for their internal technology team, as a starting point for creating their own version of the fitting room. In demonstrating my completed product, I was careful to adapt my presentation styles for the different audiences I was presenting to, including members of the public, clients and peers.

My team work in the Netherlands also included creating a game using Ogre3D. In this I applied my mathematical and programming skills in implementing a pathfinding algorithm for the in-game enemies to reach the player; this was an optimised version of the A* algorithm. I also followed a physics module and was fascinated to learn how physics engines are implemented in games. I enhanced my graphical programming skills through learning to use DirectX to manipulate existing 3D environments by adding and removing geometry in real-time, implementing a shader in HLSL that was originally written in GLSL, and creating an SSAO post-processing effect.

4th Year

During my spare time at University I wrote a program that allowed users to write and view post-processing shaders in real-time. I used this program to create a normal-map style effect, but when I tested the resulting texture, I found that it was inaccurate. As a result, for my final year project I decided to create a tool to generate functional, working normal maps for and from existing albedo textures, targeted at artists and indie developers.

- To enable easy portability to different platforms and fast running times I used C++ and OpenGL. As I believe it is important to fully understand how the underlying technologies

perform and function, I did not want to use too many external libraries. In addition, I was striving to ensure the application did not become bloated.

- The program used two windows, one with instructions the other with a preview of the normal map. The user could import a texture, then once selected, the normal map for that texture would appear in the preview window. This was generated using a fragment shader. The algorithm itself treated the input image as a height map using the average of the 3 colour channels as height values. The code I wrote implemented an existing algorithm written using the Unity Shader Graph.
- The design process was iterative - to evaluate the maps it produced I interviewed potential users about their experiences with the program and their preferences for the resulting materials, versus those that used normal maps generated by an existing program. With this feedback I made changes to my algorithm to improve the contrast in the normal maps. I did this using a technique called inverse linear interpolation.

In my final year, I applied my newly learnt skills in multithreading and concurrency when I was tasked with creating a program to find an approximation for the value of Pi. I decided to use the Monte Carlo algorithm and created two implementations on the CPU, one using OMP and the other using the standard thread library, and a third utilising the GPU. I found the GPU method to be much faster than the CPU methods and confirmed that processes to be executed in parallel would likely be best done on the GPU. I really enjoyed learning the new techniques from completing this task and they have truly enhanced my understanding of parallel processing.

Hobbies and Interests

In my free time I volunteer for an online fan-translation group, helping create translations for web-comics and manga. As an administrator I help recruit and train new members of the group. Our high-quality translations drew the attention of a Japanese Company who commissioned us to undertake some bespoke work, enabling us to upgrade our infrastructure.

I created an online reader application for the group using PHP, which has several measures in place to copyright protect our images; and after switching communication platforms from Slack to Discord, I have created bots using NodeJS to help enforce our server's rules in public channels.

Presently I am creating a C++ library for developers to create their own bots for Discord, having found that the existing ones are lacking in features and rely too heavily on external libraries.

Since leaving university I have extended my interest in new technologies, especially graphical ones, and am developing a game using Vulkan. Recently I have been using OpenCL to analyse the performance benefit from parallelising various problems – such as categorising numbers as odd or even – by doing this I have learnt how OpenCL works and how to structure programs that use it. I can now apply what I have learnt from doing this in other areas, such as writing compute shaders for OpenGL applications.

I have a passion for videogames. I enjoy playing them in my spare time, but more than that, I find enjoyment in figuring out how they are put together, and how they work. Often, I will use DLL-injection to try out my designs for post-processing shaders on existing games, my most recent one was an error diffusion style shader. These are written in a variant of HLSL.

Conclusion

Ever since I discovered graphics programming as a specialisation, it has been my goal to work in this field in a professional setting. I believe that you would provide me with an excellent opportunity to apply my current skills and develop them further in a cutting-edge environment.

References are available upon request