





# Juyong Jeong

Game programmer

 206-468-7054  jaykop.jy@gmail.com  linkedin.com/in/juyong-jeong  github.com/jaykop  juyongjeong.com

---

## Technical Skills

- Languages: C++, Python, C#, C, GLSL
  - Tools: Visual Studio, Git, OpenGL, Unity
  - Skills: Data Structure, Algorithm Analysis, Game Artificial Intelligence, 3D Graphics
- 

## Academic Projects

**Gameplay programmer - Welcome to the Future (Team of 5 Programmers)** 01/2020 - 05/2020  
2.5D side-scrolling shooting game (C#, Unity)

- Implemented a custom behavior tree structure to make design boss A.I. easy and to produce several types of them
- Designed enemy attack patterns such as flocking, shooting, throwing a grenade

**Engine Programmer - JEngine (Solo Side Project)** 09/2017- Present  
A framework to demonstrate 3D graphics and A.I. techniques (C++, OpenGL, GLSL, custom engine)

- Built the framework using builder design pattern which allows simple addition of user-written game logics
- Implemented a graphic system to demonstrate rendering 3D models, texts, particle effects, and lighting effects
- Demonstrated A.I. techniques such as agents sending messages to each other with a finite state machine and steering behaviors such as wandering, arrival, seeking, and evading

**AI Programmer - Candlelight (Team of 3 Programmers, 4 Designers)** 09/2018 - 12/2019  
3D 1st person horror game (C#, Unity)

- Developed a custom behavior tree which allows inserting and removing any behavior during runtime
- Modeled ghost A.I. using behavior tree such as detecting, chasing the player, singing, and dancing around
- Designed ghost appearance events adding activation conditions into the A.I. behavior tree

**Engine Programmer - Captain Korea (Team of 2 Programmers)** 09/2016 - 05/2017  
2D top-down view action/sneaking game (C++, OpenGL, GLSL, custom engine)

- Constructed a component-based game engine to reuse the game component
  - Designed state manager which allows modification of game states to be easy and time-efficient during the development
  - Developed a JSON parser which loads window size, fullscreen mode, and game level data
  - Built enemy A.I. tracking the player using A\* pathfinding algorithm with the predefined nodes in the game level
  - Implemented visual effects such as rippling, particle effect on the main menu using GLSL
- 

## Professional Experience

**Engine Programmer** 09/2017 - 06/2018  
DigiPen Institute of Technology (Daegu, South Korea)

- Built 2D component-based game engine for students who take a game project class
  - Developed a graphic system which supports rendering textures, animations, texts, and particle effects
  - Updated the engine with new features and bug fixes gathering feedback from students
- 

## Education

**BS in Computer Science in Real-Time Interactive Simulation** 04/2020  
DigiPen Institute of Technology (Redmond, United States)