ROGUE

Real-time Open-source Game Using Entities

Press [enter] to begin. ——————— v 0.1

R.O.G.U.E.

Real-time Open-source
Game Using Entities

MIT License

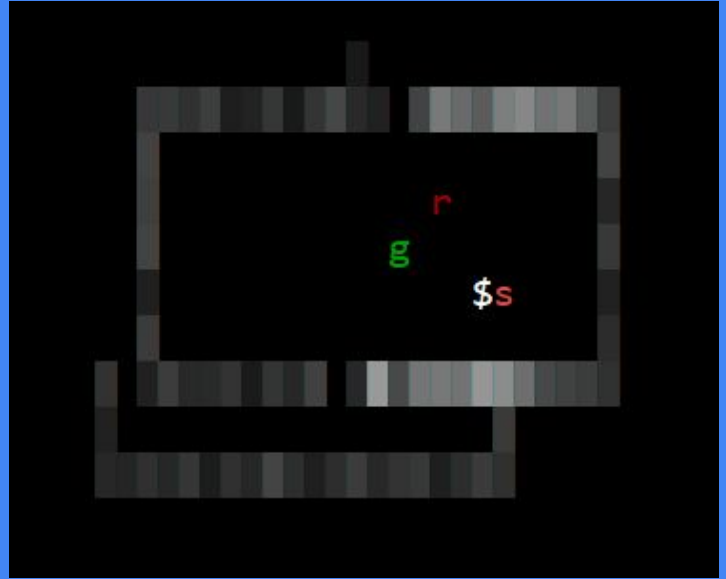Contact Info
 Daniel Bramblett <bram4@pdx.edu>
 Daniel Dupriest <kououken@gmail.com>
 Brandon Goldbeck <bgoldbeck21@gmail.com>

Target Audience

- Rogue Game Enthusiasts
    - Gamers
    - Game Makers
- Gamers in general
- Programmers looking for inspiration using:
    - .NET core
    - Entity Component System Architecture

Unique Project Features

- **Entity Component System** Architecture in C#
- **A*** Pathfinding
- Real-time Global Light Illumination
- Dungeon Generation Based on Recursive Backtracker / Flood-fill
- Vision-based Enemy Aggro using Line Tracing.
- We have **raptors** with **x-ray vision** that can **open doors**.

Other Available Projects

There are multiple options for rogue games:
- NetHack: https://www.nethack.org/
- Angband: http://rephial.org/

As well as rogue game makers.
- UnicornHack: https://github.com/AndriySvyryd/UnicornHack
- Roguelike Toolkit: https://github.com/thebracket/rltk

Project Decisions

- Using an architecture that allows features to be separated into small files (like ECS) makes collaboration much easier. Say goodbye to merge conflicts!
- Real-time vs turn-based.
- Text console vs graphics library.
  - Our game uses the .NET core to have a multi-colored UI that is cross-compatible between operating systems.
  - Can be player over ssh connection.
- We also have raptors.

Lessons Learned

- Writing the requirements before starting the project.
- Open source games in particular face design challenges relating to differing visions of how the game should look or play.
- Put more thought into how the language impacts the project. Having C# manage the memory made it harder to use the ECS architecture.

Improvements

- Manual memory management, giving more control over freeing memory. C# has unsafe code blocks where manual memory management is possible. C++ should be faster.
- Implementing a game menu.
- Fix a bit of lag on the user input.
- AI Behavior Trees.
- More dungeon types. Forest? Cave?

Contact Information

- [Github] https://www.github.com/bgoldbeck/rogue
- [Documentation] https://bgoldbeck.github.io/rogue/annotated.html
- Contact Information
  - Daniel Bramblett <bram4@pdx.edu>
  - Daniel Dupriest <kououken@gmail.com>
  - Brandon Goldbeck <bgoldbeck21@gmail.com>

Questions?