

Project: Rise of the AI

Many of the most famous games of all time had very simple AI. Let's create some classic AI behaviors.

I discuss the project more here: <https://youtu.be/z-Q3EYWyS00>

You must use new assets (art, music, sound) for this project!

You can use the font, but other than that, **you can not use anything from the course github!** (even if you did not use them on previous projects). No ctg.png, george_0.png, tileset.png, and none of the music or sound, etc.

You must find new assets.

There are some great sources for free assets:

- More by the same artist: <https://kenney.nl/assets>
- itch.io has tons of awesome free graphics: <https://itch.io/game-assets/free>
- Lots of different kinds of free assets: <https://opengameart.org/>
- Royalty free music: <https://incompetech.com/music/royalty-free/music.html>
- Again, make sure something you find **is not something in the course github**.

What are the requirements for the project?

The following are **required** to earn points for the project:

3 AI with Different Behaviors (40%)

- There needs to be 3 or more enemies with basic AI.
- Each AI should have a different behavior. For instance, one could be walking, another patrolling, another jumping or shooting. All AI must be doing something (**they can not do nothing**).
- It is OK to use the same image/textures for all 3 of your AI.

Defeat Enemies (20%)

- The player must be able to get rid of each individual enemy one at a time. You can use jumping on enemies or for a challenge, try shooting.
- Enemies that are gone should not be drawn any more or able to be collided with.

You Lose (20%)

- If the player touches an enemy, gets shot, etc. show text, "You Lose"

You Win (20%)

- If the player gets rid of all the enemies, show text "You Win"

Note on Level Layout

- Your AI can not all be on one flat floor.
- Some of your AI should be on **platforms**.
- **You will lose points** if all 3 of your AI are on the same floor.

Any tips on how to get started or approach this project?

Check collisions from the player's perspective. In other words, check if the player collided with the enemy, not if the enemy collided with the player. You may need to add code to Entity.h, Entity.cpp to make this project work.

If I want to go that extra distance, what are some things I can add?

While **not required**, here are some ideas for things to add to your game:

- If you want to really push yourself, try to implement shooting for your player and for your enemies.

I'm stuck working on my project and can't figure something out. How can I get help?

Students can reach out to me anytime: cguida@nyu.edu - If you are emailing me for help with your projects, **upload your entire project to github** and email me with the link (I need to see everything so I can help you).

Do not email screenshots of your code.

Additionally, there is a forum in NYU Classes where **students can help each other**.

How do I submit my work?

Commit your code to your GitHub repository. Post the link to your github in the **Assignments** area in NYU Classes. For example, your link might look like:

<https://github.com/tonystark/CS3113/P1/>

If you are having difficulties with github, you can **.zip your entire project** and post a link to google drive. Make sure your google drive link is **accessible to everyone** so that myself and the TA can access it.

Due by 11:59pm means your project was successfully uploaded and a link was submitted to NYU Classes by that time. Start uploading your project at least an hour before the deadline. **Projects received 1 minute late are considered to be a day late.**

If there are any issues with uploading your project, you must **email me before the due date.**

While I check email regularly, **do not expect a response over the weekend or close to deadlines.**

Your code must compile. Code that does not compile will receive a grade of 0.

Late projects will have **10 points deducted per day**. Late projects will **not be accepted after 2 days**.