

Spring 2025 - CSC329 Game Programming - Assignment 2



Due date: Thursday, February 6, 2025, 11 am, add and commit a folder assignment2/ to our class SVN repository.

Exercise 2.1 [4 points]

This exercise contains review questions. The answers should be given in a **plain text file** named `answers.txt`. Please make sure the filename matches this exactly and resides in the `assignment2/` directory.

- What is the thread of control in a program?
[2 points]
- What kind of tasks do not require any kind of synchronization?
[2 points]

Exercise 2.2 [16 points]

This is a programming exercise. We expect you to (1) build your project as a Unity game and commit both the (2) **sources** and the (3) **binary** to the class SVN-Repository (instructions below). The sources should be placed in the folder `assignment2/src` whereas the binary should be placed in `assignment2/bin`. Create these directories now. Get the latest stable recommended version of Unity (e.g. LTS Release 6000.0.33f1 for Mac Silicon – the version might be different for other platforms). If you are using Unity-Hub you need to change the platform in the Unity-Hub application.

1. Creating your Unity game

- (a) Download Unity from the appropriate web page, and create a new 2D Core project. Import or create assets (at least one background and a few sprites), compile and run the game.
[7 points]
- (b) Create your own 2D graphics with your program of choice. To use animations, you can use both options that we discussed in class: (a) creating images (at least five), loading them into your project, and creating an animation loop to cycle your images, or (b) using a sprite sheet with various parts that can be animated independently. You need to get familiar with the Unity 2D animation tools in both cases. Appropriate speeds for image cycling and image movement must be selected so that the result is visually pleasing.
[9 points]

2. Create your sources (as Unity package file)

- (a) After you have completed your project in the Unity Client you will need to create a package file. Go to `Assets→Export Package...` and in the window that pops up select

all the items you want to export. Then press the 'Export...' button and navigate your package to your assignment2/src folder. Name the export after your game project.

3. Create your binary (as Unity WebGL)

- (a) After you have completed your project in the Unity Client you will need to ensure the project is built for the web. Go to File→Build Profile... and in the window that pops up select Web under Platform. Click on the button "Switch Platform". Unity will switch platforms, this might take a few seconds. Be patient.
- (b) Remove compression format "Brotli/Gzip": Go to Edit → Project Settings → Player → Publishing Settings for the Web platform. In the Compression Format dropdown, select Disabled. This will turn off compression for your WebGL build
- (c) Change "Player settings". You need to change some settings so that the WebGL scene can be opened with a web server. Select "Player" in the left pane (should be selected by default) and expand "Publishing Settings" in the right pane. The second line shows the Compression Format. Set it to "Disabled".
- (d) After you have configured the Web platform, click "Build". A dialog will pop up and ask you to save the build. Select your assignment2/bin folder and name the export after your game project. This might take some time, especially the first time.
- (e) Once done, run a web server to check if the binary behaves the way it should. I recommend to cd into your assignment2/bin folder and then cd into the folder you just created when saving your game. Inside that folder is an index.html file. Inside this folder run a web server, e.g. `python3 -m http.server 8888`. Then open a browser and go to `http://localhost:8888`. Your game should start.

4. **Commit assignment2 to the SVN repository** Now add and commit the entire assignment 2 to our class SVN repository. If you have trouble: check our [class website](#) for details on how to do this.