

## [GAM-120-JJ1: Intro to Game Logic – Spring 2025](#)

### Course Syllabus

---

**Meeting Time and Location:** Friday, 1:10-4:10 PM, Fine Arts 249

**Instructor:** Matthew DiMatteo (he/him)

**Email:** [mdimatteo@rider.edu](mailto:mdimatteo@rider.edu)

**Office Hours:** Monday and Tuesday 5:30-6:30, Friday 4:30-5:30 or by appointment, Fine Arts 214

*If I'm not in my office, check for me in our classroom (Fine Arts 249)*

---

### Contents

Overview .....	2
Materials .....	3
Assignments and Grading.....	4
Course Schedule .....	6
Assignment Instructions .....	10
Academic Policies .....	18

---

Mutual respect and a commitment to inclusiveness are crucial to a positive learning environment. In this course, we will honor the diversity of all members of the Rider community by fostering a learning environment that is respectful of other classmates based on their identities and past experiences, including race, ethnicity, national origin, gender, sexuality, age, religion, culture, veteran status, and disability. I encourage any student who has concerns about the climate of this classroom or the behavior of others in the class to discuss matters with the instructor or the chair of the instructor's department.

---

### Health and Wellness Resources

**[Student Health Center](#):** Poyda Hall – [healthcenter@rider.edu](mailto:healthcenter@rider.edu) **609-896-5060**

**[Counseling Center](#):** Zoerner House – [counseling@rider.edu](mailto:counseling@rider.edu) **609-896-5157**

**[Healthy Broncs Portal](#)** – Appointments, Medical Forms

**National Suicide and Crisis Lifeline:** **Dial 988**

**Local Therapy Resources:** <https://findtreatment.samhsa.gov/>

**[Report an Incident](#)**

---

### Tutoring and Accessibility Support

**[Academic Success Center](#):** Bart Luedeke Center, Suite 237, [academicsuccesscenter@rider.edu](mailto:academicsuccesscenter@rider.edu)

**[Student Accessibility and Support Services](#):** Bart Luedeke Center, Suite 201, [accessibility@rider.edu](mailto:accessibility@rider.edu)

## Course Description

[Intro to Game Logic](#) introduces students to strategies for technical implementation of digital games. Students learn programming techniques and design patterns for popular genres and formats through small game development projects and problem-solving exercises. Weekly lessons include a mix of technical demonstration and lab time for help and troubleshooting.

---

## Course Learning Goals

Students will:

1. Understand the logic governing the construction of digital games.
  2. Understand traditional design patterns for constructing different forms of digital games.
  3. Understand foundational programming concepts transferable to other languages and environments.
  4. Understand techniques for constructing game worlds in various formats.
  5. Construct, test, and revise digital game prototypes.
  6. Understand techniques for troubleshooting digital games.
  7. Leverage online learning resources to become independent problem solvers.
- 

## Course Requirements

1. Students will be expected to arrive to class on time, follow along with in-class demos, participate in workshops, and complete assignments by the date due. In the event of absence, students are responsible for communicating with the instructor in a timely manner, catching up on material covered in class, and completing any missed assignments.
2. Students should expect to spend a few hours per week outside of class time to complete assignments. Students are strongly encouraged to work consistently throughout the semester. Always take into account lab hours and possible technical problems when planning the time you will spend on assignments.
3. Students are expected to make use of learning resources available online in addition to any aid provided in the classroom.
4. Students are responsible for saving and backing up their work, and are strongly encouraged to utilize multiple backup locations, such as external hard drives and cloud storage services (such as Google Drive, Dropbox, etc.) in addition to personal computers. Because students are expected to routinely back up their files, the loss of data is not considered an acceptable excuse for late or missing work.

---

## Course Materials

- **Canvas** – Resources such as [class slides](#), [code examples](#), and other materials will be posted in the [Files](#) section. Assignment instructions and submission can be found in the [Assignments](#) section. Quick links and a week-by-week schedule can be found in the [Modules](#) section. The [Announcements](#) section will be used to post notifications on weather-related scheduling changes or any other news relevant to the class. Any announcements made will also be copied as a class-wide email.
- **PICO-8** – Students can use the free, educational version of the PICO-8 game engine, which runs in a web browser: <https://www.pico-8-edu.com/>. The desktop version of the software is also available, though only for Windows. [Download it from Canvas](#).
- **GitHub** – I’ve created a GitHub repository called “[helloworld](#)” that contains several sets of examples for the types of projects we’ll be completing this semester. [Download the latest release](#)
- **PICO-8 Learning Resources** – There is no textbook for this course. In addition to the GitHub examples, students are strongly recommended to refer to PICO-8’s online documentation and learning resources:
  - [PICO-8 Home](#)
  - [Documentation and Tutorials](#)
  - [Cheat Sheet](#) (Quick Reference)
  - [PICO-8 Wiki](#) (Reference for PICO-8 Functions)
  - [More](#) (Download PDF from Canvas)
- **Optional: [Microsoft Visual Studio Code](#) (or alternative programming environment)**
  - PICO-8’s code editor utilizes a heavily stylized font and runs in a relatively small window; some students may prefer to do their coding in an external editor. Microsoft Visual Studio Code is an industry-standard, cross-platform tool that can be downloaded for free and installed on a personal computer. It is also installed on each of the workstations in our classroom.
  - Highly recommended is the [pico8-ls extension](#) which provides syntax highlighting for the Lua programming language which PICO-8 runs on. Setup and workflow for using an external code editor will be covered in the first few weeks of class.

## Assignments by Category

<b>Projects (3 for 15% each)</b>		<b>45%</b>
• Due Week 06 Feb. 28	<a href="#">Project #1: Paddleball Game</a>	15%
• Due Week 10 Apr. 11	<a href="#">Project #2: Top-Down Adventure Game</a>	15%
• Due at Finals May 9	<a href="#">Project #3: Sidescrolling Platformer Game</a>	15%
<b>Exercise Sets (3 for 10% each)</b>		<b>30%</b>
• Due Week 05 Feb. 21	<a href="#">Exercise Set #1</a>	10%
• Due Week 09 Apr. 4	<a href="#">Exercise Set #2</a>	10%
• Due Week 13 May 2	<a href="#">Exercise Set #3</a>	10%
<b>Progress Check-ins (3 for 5% each)</b>		<b>15%</b>
• Due Week 04 Feb. 14	<a href="#">Progress Check-in #1</a>	5%
• Due Week 08 Mar. 28	<a href="#">Progress Check-in #2</a>	5%
• Due Week 12 Apr. 25	<a href="#">Progress Check-in #3</a>	5%
<b>Attendance and Participation</b>		<b>10%</b>

## Assignments Calendar

• Due Week 04 Feb. 21	<a href="#">Progress Check-in #1</a>	5%
• Due Week 05 Feb. 28	<a href="#">Exercise Set #1</a>	10%
• Due Week 06 Mar. 7	<a href="#">Project #1: Paddleball Game</a>	15%
• Due Week 08 Mar. 28	<a href="#">Progress Check-in #2</a>	5%
• Due Week 09 Apr. 4	<a href="#">Exercise Set #2</a>	10%
• Due Week 10 Apr. 11	<a href="#">Project #2: Top-Down Adventure Game</a>	15%
• Due Week 12 Apr. 25	<a href="#">Progress Check-in #3</a>	5%
• Due Week 13 May 2	<a href="#">Exercise Set #3</a>	10%
• Due at Finals May 9	<a href="#">Project #3: Sidescrolling Platformer Game</a>	15%
• Assessed at semester's end:	<a href="#">Attendance and Participation</a>	10%

\* See pages 10-17 for detailed assignment instructions, or visit the [Assignments](#) section on Canvas

---

## Grading Scale

A	93.50-100%
A-	89.50-93.49%
B+	86.50-89.49%
B	83.50-86.59%
B-	79.50-83.49%
C+	76.50-79.49%
C	73.50-76.49%
C-	69.50-73.49%
D	59.50-69.49%
F	0-59.49%

Students must earn a **D or higher to pass the course** and earn its 3 credits

Students must average a **C or higher across their courses** (2.0 GPA) to [remain in Good Academic Standing](#)

---

## Late Work

- All assignments (unless otherwise specified) must be submitted to [Canvas](#) by **5:30 PM** on the date due.
- Assignments turned in **less than one week late** will be subject to a **5% grade penalty**.
- Assignments turned in **more than one week late** will be subject to a **10% grade penalty**.
- Students with **more than one missing assignment** will be subject to an **additional 10% grade penalty** on late assignments each week they are late.
- For students with **only one missing assignment**, the **grade penalty will be capped at 10%**, meaning students can still receive up to 90% credit on that late assignment as long as it is submitted by the end of the semester.
  - My goal is to give students every opportunity to benefit from coursework. *Please note that late work policies will likely differ in your other courses and are determined by individual instructors.*
- Assignments not turned in at all will receive a grade of 0.
- Students will have **2 “free passes”** for requesting **extensions on assignments**. *Extensions may not be longer than one week.*

# Course Schedule (subject to change)

---

## Week 01 (Jan. 31) — Course Overview, Getting Started with PICO-8

- Introductions, Course Expectations, Materials, Assignments and Grading, Student Support Resources
  - Example PICO-8 Games
  - Getting Started with the [PICO-8 Editor](#): Using the Editor, Drawing Sprites, Saving Your Work
  - Displaying Text with [print\(\)](#): [basics\\_00\\_helloworld.p8](#)
  - Drawing Shapes: [basics\\_01\\_shapes.p8](#) | [rect\(\)](#), [rectfill\(\)](#), [circ\(\)](#), [circfill\(\)](#)
  - The PICO-8 Coordinate System: [basics\\_04\\_coordinates.p8](#)
  - Displaying Sprites Using [spr\(\)](#): [paddleball\\_basic\\_01\\_sprites.p8](#)
  - [GitHub Repository for this Course: helloworld](#) | [Download Latest Release](#) | [PICO-8 Learning Resources](#)
  - [Optional Student Stress Factors Survey](#)
- 

## Week 02 (Feb. 7) — Creating a Paddleball Game: Variables

- The [PICO-8 Game Loop](#): [\\_init\(\)](#), [\\_update\(\)](#), [\\_draw\(\)](#) | [new\\_gameloop.p8](#)
  - Using Variables to Move a Ball Across the Screen: [paddleball\\_basic\\_02\\_variables.p8](#)
  - Using Variables for Different Objects (Paddle and Ball): [paddleball\\_basic\\_03\\_variables2.p8](#)
  - Setting up an External Code Editor: [Microsoft Visual Studio Code](#), [pico8-ls Extension](#)
  - **Upcoming:**
    - [Progress Check-in #1](#) (Due Week 4, Feb. 21)
    - [Exercise Set #1](#) (Due Week 5, Feb. 28)
    - [Project #1: Paddleball Game](#) (Due Week 6, Mar. 7)
- 

## Week 03 (Feb. 14) — Creating a Paddleball Game: Functions and Conditional Statements

- Handling Input: [paddleball\\_basic\\_04\\_input\\_and\\_movement.p8](#) | [btn\(\)](#), [btnp\(\)](#)
  - Keeping the Paddle on Screen Using If/Then Statements: [paddleball\\_basic\\_05\\_conditional\\_logic.p8](#)
    - Comparison Operators `<` `<=` `>` `>=` `==`
  - Writing and “Calling” Your Own Functions: [paddleball\\_basic\\_06\\_functions.p8](#)
  - Organizing Your Code Using Tabs (in the PICO-8 Editor)
  - **Due Next Week:** [Progress Check-in #1](#) (50 pts / 5% of Semester Grade)
  - **Upcoming:**
    - [Exercise Set #1](#) (Due Week 5, Feb. 28)
    - [Project #1: Paddleball Game](#) (Due Week 6, Mar. 7)
- 

## Week 04 (Feb. 21) — Creating a Paddleball Game: Conditional Logic

- Detecting When Two Objects (Ball and Paddle) Intersect: [paddleball\\_basic\\_07\\_collision.p8](#)
  - [Download Diagram from Canvas](#) | [Truth Tables Spreadsheet Example](#)
  - Logical Operators AND, OR
- Making the Ball Bounce When it Hits the Paddle: [paddleball\\_basic\\_08\\_reverse\\_direction.p8](#)
- Moving the Ball Sideways and Bouncing Off the Walls and Ceiling: [paddleball\\_basic\\_09\\_bounce.p8](#)

- Updating Variables; Resetting the Ball When Missed: [paddleball\\_basic\\_10\\_score\\_and\\_lives.p8](#)
  - **Due by 5:30 PM:** [Progress Check-in #1](#) (50 pts / 5% of Semester Grade)
  - **Due Next Week:** [Exercise Set #1](#) (100 pts / 10% of Semester Grade)
  - **Upcoming:** [Project #1: Paddleball Game](#) (Due Week 6, Mar. 7)
- 

### **Week 05 (Feb. 28) — Paddleball Game Workshop: Polish & Wrap-up**

- Using the Sound Editor and Adding Sounds
  - Ending and Restarting the Game: [paddleball\\_basic\\_11\\_gameover.p8](#)
  - Demos on Demand, Lab Time & Questions, Help & Troubleshooting
  - Download all Paddleball Step-by-Step Examples: [\\_helloworld\\_01\\_paddleball\\_basic.zip](#)
  - Optional: Try an Advanced Version using Physics: [\\_helloworld\\_01a\\_paddleball\\_physics.zip](#)
  - **Due by 5:30 PM:** [Exercise Set #1](#) (100 pts / 10% of Semester Grade)
  - **Due Next Week:** [Project #1: Paddleball Game](#) (Due Week 6, Mar. 7)
- 

### **Week 06 (Mar. 7) — Paddleball Game Workshop: Help & Troubleshooting**

- Demos on Demand, Lab Time & Questions, Help & Troubleshooting
  - **Due by 5:30 PM:** [Project #1: Paddleball Game](#) (150 pts / 15% of Semester Grade)
  - **Due Next Week:** [Mid-Semester Student Feedback Survey](#) (Optional, Ungraded)
- 

### **Week 07 (Mar. 14) — Creating a Top-Down Adventure Game: Tile-Based Movement**

- Using the Map Editor | Tile Coordinates vs. Pixel Coordinates
  - Getting Started: [topdown\\_00\\_assets\\_only.p8](#)
  - Using Tables to Store Lists of Values: [tables.p8](#)
  - Using Tables for Game Objects: [object\\_tables.p8](#) | [Tables Spreadsheet Example](#)
  - Tile-Based Movement: [topdown\\_01\\_tilebased\\_movement\\_and\\_collision.p8](#)
    - Sprite Flags and Collision | [mget\(\)](#), [fget\(\)](#)
  - [Top-Down Adventure Tutorial Playlist by Dylan Bennett](#)
  - **Due by 5:30 PM:** [Mid-Semester Student Feedback Survey](#) (Optional, Ungraded)
  - **Due Next Class:** [Progress Check-in #2](#) (50 pts / 5% of Semester Grade)
  - **Upcoming:**
    - [Exercise Set #2](#) (Due Week 9, Apr. 4)
    - [Project #2: Top-Down Adventure Game](#) (Due Week 10, Apr. 11)
- 

*No Class Mar. 21 (Spring Break)*

---

---

## Week 08 (Mar. 28) — Creating a Top-Down Adventure Game: Interacting with Tiles

- Collecting Items / Interacting with Map Tiles: [topdown\\_02\\_item\\_collection.p8](#) | [mset\(\)](#)
- Lock & Key Systems: [topdown03\\_lock\\_and\\_key.p8](#)
- [Top-Down Adventure Tutorial Playlist by Dylan Bennett](#)
- **Due by 5:30 PM:** [Progress Check-in #2](#) (50 pts / 5% of Semester Grade)
- **Due Next Week:** [Exercise Set #2](#) (100 pts / 10% of Semester Grade)
- **Upcoming:** [Project #2: Top-Down Adventure Game](#) (Due Week 10, Apr. 11)

---

## Week 09 (Apr. 4) — Creating a Top-Down Adventure Game: Camera and Map

- Expanding on Item Collection and Lock & Key Systems
- Positioning the Camera: [topdown04\\_camera\\_modes.p8](#)
- Dungeons and Interior Areas, Warping: [topdown05\\_warping.p8](#)
- [Top-Down Adventure Tutorial Playlist by Dylan Bennett](#)
- **Due by 5:30 PM:** [Exercise Set #2](#) (100 pts / 10% of Semester Grade)
- **Due Next Week:** [Project #2: Top-Down Adventure Game](#) (150 pts / 15% of Semester Grade)

---

## Week 10 (Apr. 11) — Top-Down Adventure Game Workshop: Help & Troubleshooting

- Demos on Demand, Lab Time & Questions, Help & Troubleshooting
- Download all Top-Down Adventure Step-by-Step Examples: [\\_helloworld\\_02\\_topdown\\_adventure.zip](#)
- [Top-Down Adventure Tutorial Playlist by Dylan Bennett](#)
- **Due by 5:30 PM:** [Project #2: Top-Down Adventure Game](#) (150 pts / 15% of Semester Grade)

*If you wish to [withdraw from a course](#), you must do so by Apr. 11 ([see Academic Calendar](#))*

---

## Week 11 (Apr. 18) — Creating a Sidescrolling Platformer Game: Movement and Collision

- Getting Started: [platformer\\_00\\_assets\\_only.p8](#) | Positioning the Player: [platformer\\_01\\_starting\\_position.p8](#)
- Moving the Player Horizontally: [platformer\\_02\\_horizontal\\_movement.p8](#)
- Delta Time: [platformer\\_03\\_delta\\_x.p8](#) | [Delta Time Spreadsheet Example](#)
- Physics-Based Movement and Friction: [platformer\\_04\\_friction.p8](#)
- Jumping and Gravity: [platformer\\_05\\_jumping.p8](#) | [platformer\\_06\\_gravity.p8](#)
- Map Collision: [platformer\\_07\\_map\\_collision.p8](#) | Adjusting the Jump Feature: [platformer\\_08\\_jumping2.p8](#)
- Horizontal Map Collision: [platformer\\_09\\_walls.p8](#)
- [Side-Scrolling Platformer Tutorial Playlist by Nerdy Teachers on YouTube](#)
- **Due Next Week:** [Progress Check-in #3](#) (50 pts / 5% of Semester Grade)
- **Upcoming:**
  - [Exercise Set #3](#) (Due Week 13, May 2)
  - [Project #3: Sidescrolling Platformer Game](#) (Due at Finals, May 9)



---

## Week 12 (Apr. 25) — Creating a Sidescrolling Platformer Game: Additional Features

- Refining Movement: [platformer\\_10\\_movement\\_refined.p8](#) | [platformer\\_10a\\_correction\\_illustrated.p8](#)
- Positioning the Camera: [platformer\\_11\\_camera.p8](#)
- Pits and Respawning: [platformer\\_12\\_pits\\_and\\_respawn.p8](#)
- Animating a Sprite: [platformer\\_13\\_animated\\_sprite.p8](#) | [platformer\\_13a\\_toggle\\_animation.p8](#)
- Switching Between Levels: [platformer\\_14\\_multiple\\_levels.p8](#)
- Friction and Special Terrain: [platformer\\_15\\_terrain\\_and\\_friction.p8](#)
- [Side-Scrolling Platformer Tutorial Playlist by Nerdy Teachers on YouTube](#)
- **Due by 5:30 PM:** [Progress Check-in #3](#) (50 pts / 5% of Semester Grade)
- **Due Next Week:** [Exercise Set #3](#) (100 pts / 10% of Semester Grade)
- **Upcoming:** [Project #3: Sidescrolling Platformer Game](#) (Due at Finals, May 9)

*Course Evaluations will be open near the end of the semester (an email will be sent with the precise dates) – if at least 75% of the class completes these, I will give everyone 1 percentage point **extra credit** added to your final grade*

---

## Week 13 (May 2) — Sidescrolling Platformer Game Workshop: Help & Troubleshooting

- Demos on Demand, Lab Time & Questions, Help & Troubleshooting
- Download all Platformer Step-by-Step Examples: [\\_helloworld\\_03\\_sidescrolling\\_platformer.zip](#)
- Optional: Projectiles, Pickups, and Enemies (Advanced): [\\_helloworld\\_04\\_space\\_shooter.zip](#)
- Optional: Timers, Game States, Misc Features: [\\_helloworld\\_05\\_misc\\_features.zip](#)
- [Side-Scrolling Platformer Tutorial Playlist by Nerdy Teachers on YouTube](#)
- **Due by 5:30 PM:** [Exercise Set #3](#) (100 pts / 10% of Semester Grade)
- **Due Next Week:** [Project #3: Sidescrolling Platformer Game](#) (150 pts / 15% of Semester Grade)

*If you wish to request a grade of [Incomplete](#) for this course, you must do so by this date*

---

## Final Exam Period (May 9) — Sidescrolling Platformer Game Workshop: Help & Troubleshooting

**Final Exam Meeting:** **Friday, May 9, 1:00-3:00 PM, FA 249 (our usual location, slightly earlier)**

- Lab Time & Questions, Help & Troubleshooting
  - **Due by 5:30 PM:** [Project #3: Sidescrolling Platformer Game](#) (150 pts / 15% of Semester Grade)
- 

## Closing the Semester

Assessed at end of semester: [Attendance and Participation](#) (100 pts / 10% of Semester Grade)

*Canvas Course closes **Tue. May 13 at 12:00 PM** – submit any missing work by this deadline*

# Assignment Instructions

---

## Progress Check-ins

Due [Week 4 \(Feb. 21\)](#), [Week 8 \(Mar. 28\)](#), [Week 12 \(Apr. 25\)](#)

50 pts (5% of Semester Grade) Each

## Instructions

Because not everyone is vocal in class, it can be difficult for me to assess how everyone is handling the material between project due dates. These check-ins are intended to provide an **opportunity for you to voice any questions or concerns** about the material, your performance in class, or the course content, and to **help me understand your level of comfort and familiarity** with the concepts being covered in class, so I can better help you learn.

## Requirements

- Please **upload your PICO-8 file** (.p8 format) for the current project to **share your code** with me.
- In **writing**, **OR as an audio recording**, please provide a **summary of your progress**:
  - What is giving you the most **trouble**? What **questions** do you have?
  - What are you most **proud** of?
  - What are you doing **next**?

## Submitting Your Work

- **Save your game** as a **.p8** file. You can do this in the PICO-8 editor by using the command **SAVE**
- **Upload your .p8 file** to the corresponding [Assignments](#) page on Canvas.
- **Upload your summary** of progress as a Google Doc, Word Doc, PDF, text file, text entry, or audio recording.
- Submit all materials by **5:30 PM** on the due date listed above.

## Grading Rubric *(students will receive a percentage of the point total above, based on the quality of their work)*

- **100%** The PICO-8 (.p8) file to run the game has been provided; all required points are addressed
- **90%** In-class participation has demonstrated good progress, but no submission has been provided
- **80%** One of the requirements above has not been adequately addressed
- **60%** Two of the requirements above have not been adequately addressed
- **30%** Three of the requirements above have not been adequately addressed
- **10%** None of the requirements above have been adequately addressed
- **0%** Nothing has been submitted, and progress in class has not been evident

## Exercise Set #1

Due Week 5 (Feb. 28)

100 pts (10% of Semester Grade)

### Instructions

Please complete **each** of the **3** following exercises. [Download the corresponding .p8 file for each exercise](#) and read the instructions at the top of the file. Each file has **something wrong with it that you must fix**, or must be modified in some way to produce alternative behavior. When you are done, save each modified file and upload to [Canvas](#) (see “Submitting Your Work” below for more details).

### Exercise 1-1: Variables

This example uses variables to determine the ball's x,y position and sprite.

- Check the sprites tab. See how there are two different sprites for the ball (blue and pink)? The ball is currently blue.
- **Goal: Modify this example so that the ball uses the pink sprite instead.**

### Exercise 1-2: Variables 2

- This example uses variables to determine the ball's x,y position and sprite.
- The ball is not moving. Why not? Make sure variables are being used in all the right places.
- **Goal: Fix this example so that the ball moves down the screen.**

### Exercise 1-3: Conditional Logic

- This example uses if/then statements to keep the paddle from going off screen.
- **Goal: Modify the example so that the paddle instead wraps around the screen**, appearing on the left side after going off screen on the right, and vice-versa (like in Pac-Man).

### Submitting Your Work

- **Save all modified examples** as **.p8** files. You can do this in the PICO-8 editor by using the command **SAVE** followed by a filename and pressing the Enter or Return key.
- **Upload your .p8 files** to [Canvas](#) by **5:30 PM** on the due date listed above.

**Grading Rubric for Exercise Sets 1-3** (students will receive a percentage of the point total above, based on the quality of their work)

- **100%** Each of the 3 files has been fixed or modified correctly according to the instructions to produce the desired effect(s) precisely
- **90%** Each file has been modified to mostly produce the desired effect(s), but one file does not match the intended behavior precisely
- **80%** Each file has been modified to mostly produce the desired effect(s), but two files do not match the intended behavior precisely
- **70%** Each file has been modified to mostly produce the desired effect(s), but all three files do not match

the intended behavior precisely; OR two files have been modified correctly, but one has not been modified correctly

- **35%** One of the files has been modified correctly, but the other two have not
- **0%** No work has been submitted

## **Project #1: Paddleball Game**

Due Week 6 (Mar. 7)

150 pts (15% of Semester Grade)

Using [PICO-8](#), create a paddleball game **expanding on** the in-class demo:

- Consider **changing properties** such as gravity and the speed and size of the paddle and ball.
- **Add one new feature** such as a second ball, blocks to destroy, powerups that change properties of game objects, hazards, alternate ways of scoring, etc.
- **Test your game** to make sure it runs! Points will be lost for game-breaking bugs.
- **Organize your code** so that it's easy to read. Name variables and functions clearly. Use consistent indentation. Add comments where helpful.
- Using the PICO-8 editor, **create original sprites for all game objects.**

### **Submitting Your Work**

- **Save your game** as a **.p8** file. You can do this in the PICO-8 editor by using the command **SAVE** followed by a filename and pressing the Enter or Return key.
- **Upload your .p8 file** to [Canvas](#) by **5:30 PM** on the due date listed above.

**Grading Rubric** (*students will receive a percentage of the point total above, based on the quality of their work*)

- **100%** There has been substantial feature innovation on top of the basic example  
The game is functional and does not crash at any point  
Code organization is excellent in terms of indentation and comments
- **90%** There has been notable feature innovation on top of the basic example  
The game is mostly functional but crashes in certain situations  
Code could be better organized in terms of indentation and comments
- **75%** There has been only modest innovation attempted
- **50%** The game requires substantial troubleshooting to run, or offers very little innovation
- **25%** The game is almost entirely nonfunctional or offers almost no innovation on top of the basic example
- **0%** No work has been submitted

---

## **Exercise Set #2**

Due Week 9 (Apr. 4)

100 pts (10% of Semester Grade)

### **Instructions**

Please complete **each** of the **3** following exercises. **Download the corresponding .p8 file for each exercise** and read the instructions at the top of the file. Each file has **something wrong with it that you must fix**, or must be modified in some way to produce alternative behavior. When you are done, save each modified file and upload to [Canvas](#) (see “Submitting Your Work” below for more details).

### **Exercise 2-1: Object Tables**

- This example shows a new way to define the properties of a game object, such as its sprite number and x,y coordinates. See how the paddle is being defined where *pad* is a table (like a spreadsheet) that can be used to contain a list of values. To add those values to the table, you type the table name (such as *pad*) followed by a dot, then whatever you want to name the variables for those values. For example, *pad.n* is the sprite number for the paddle.
- **Goal: Modify this example so that the ball is also defined as a table.** Make sure to type the ball's variable names in the same pattern as the paddle's. Make sure the ball is being displayed properly when the game runs.
- *Hint: You'll need to change the ball's variable names in two places: where the variables are first defined, and then where they are used to draw the ball.*

### **Exercise 2-2: Tile-Based Movement**

- This example is not working properly. The player is not moving as expected. The player should not be able to move through solid tiles like trees, but should be able to move otherwise.
- **Goal: Fix this example so the player moves one tile at a time and can move freely unless blocked by a solid tile like a tree or rock.**
- *Hint: How many pixels wide and tall is one tile?*

### **Exercise 2-3: Item Collection**

- This example allows the player to collect coins by walking over them. There are two types of coins: silver and gold. But right now, the player can only collect the silver coins.
- **Goal: Modify this example so the player also can collect the gold coins by walking over them.** The coins should disappear after being collected, and the player should get a greater amount of money when collecting a gold coin (silver are worth 1).

### **Submitting Your Work**

- **Save all modified examples** as **.p8** files. You can do this in the PICO-8 editor by using the command **SAVE** followed by a filename and pressing the Enter or Return key.
- **Upload your .p8 files** to [Canvas](#) by **5:30 PM** on the due date listed above.

## Project #2: Top-Down Adventure Game

Due Week 10 (Apr. 11)

150 pts (15% of Semester Grade)

Using [PICO-8](#), create a top-down adventure game **expanding on** the in-class demo:

- **Add one new feature** such as hazards, a dungeon or interior environment, timed collection, obstacles with item-based progression, etc.
- Create an **entirely original map**, different from the one in the example. Avoid large empty areas. Apply level design principles as discussed in class.
- **Test your game** to make sure it runs! Points will be lost for game-breaking bugs.
- **Organize your code** so that it's easy to read. Name variables and functions clearly. Use consistent indentation. Add comments where helpful.
- Using the PICO-8 editor, **create original sprites for all game objects**.

### Submitting Your Work

- **Save your game** as a **.p8** file. You can do this in the PICO-8 editor by using the command **SAVE** followed by a filename and pressing the Enter or Return key.
- **Upload your .p8 file** to [Canvas](#) by **5:30 PM** on the due date listed above.

**Grading Rubric** (*students will receive a percentage of the point total above, based on the quality of their work*)

- **100%** There has been substantial feature innovation on top of the basic example  
The game is functional and does not crash at any point  
Code organization is excellent in terms of indentation and comments
- **90%** There has been notable feature innovation on top of the basic example  
The game is mostly functional but crashes in certain situations  
Code could be better organized in terms of indentation and comments
- **75%** There has been only modest innovation attempted
- **50%** The game requires substantial troubleshooting to run, or offers very little innovation
- **25%** The game is almost entirely nonfunctional or offers almost no innovation on top of the basic example
- **0%** No work has been submitted

## Exercise Set #3

Due Week 13 (May 2)

100 pts (10% of Semester Grade)

### Instructions

Please complete **each** of the **3** following exercises. [Download the corresponding .p8 file for each exercise](#) and read the instructions at the top of the file. Each file has **something wrong with it that you must fix**, or must be modified in some way to produce alternative behavior. When you are done, save each modified file and upload to [Canvas](#) (see “Submitting Your Work” below for more details).

### Exercise 3-1: Ceilings

- This example allows the player to move left/right and jump. There is collision detection with the floor, so the player won't fall through the level. But there is no collision with the ceiling.
- **Goal: Modify this example so the player CANNOT jump up through the BLOCKS from below but so they CAN still jump up through the CLOUDS from below.**
- *Hint: You may want to use different sprite flags to distinguish between blocks and clouds.*

### Exercise 3-2: Hazards

- This example returns the player to the start and subtracts a life when they fall down a pit.
- **Goal:**
  - **Make the spikes and lava damage the player but not take away an entire life.**
  - **The player should still be returned to the start after landing on these hazards.**
  - **If the player's health runs out, a life should be lost and health reset.**
- *Hint: You may need to introduce a new sprite flag to represent hazard tiles, as well as add a way to keep track of health in addition to lives.*

### Exercise 3-3: Camera Pan

- This example modifies the controls so that the player can press the z key along with an arrow key to pan the camera instead of moving the player.
- Or, well, it would. it's all set up to receive those button inputs, but it's up to you to make the camera work happen.
- **Goal: Modify the example to pan the camera slightly ahead and behind the player depending on which arrow key is held down (you only need to handle left and right, not up and down).**

### Submitting Your Work

- **Save all modified examples as .p8 files.** You can do this in the PICO-8 editor by using the command **SAVE** followed by a filename and pressing the Enter or Return key.
- **Upload your .p8 files to [Canvas](#) by 5:30 PM** on the due date listed above.

## Project #3: Sidescrolling Platformer Game

Due at [Finals](#) (Friday, May 9 at 1:00 PM)

150 pts (15% of Semester Grade)

Using [PICO-8](#), create a sidescrolling platformer game **expanding on** the in-class demo:

- Consider **changing properties** such as gravity and the speed and size of the player.
- **Add one new feature** such as a powerup that changes the player's speed or makes them invincible, timed completion, a collection requirement, original enemy behavior or hazards, special terrain that changes how the player moves, attack projectiles, etc.
- Create an **entirely original map**, different from the one in the example. Avoid large empty areas. Apply level design principles as discussed in class.
- **Test your game** to make sure it runs! Points will be lost for game-breaking bugs.
- **Organize your code** so that it's easy to read. Name variables and functions clearly. Use consistent indentation. Add comments where helpful.
- Using the PICO-8 editor, **create original sprites for all game objects**.

### Submitting Your Work

- **Save your game** as a **.p8** file. You can do this in the PICO-8 editor by using the command **SAVE** followed by a filename and pressing the Enter or Return key.
- **Upload your .p8 file** to [Canvas](#) by **5:30 PM** on the due date listed above.

**Grading Rubric** (*students will receive a percentage of the point total above, based on the quality of their work*)

- **100%** There has been substantial feature innovation on top of the basic example  
The game is functional and does not crash at any point  
Code organization is excellent in terms of indentation and comments
- **90%** There has been notable feature innovation on top of the basic example  
The game is mostly functional but crashes in certain situations  
Code could be better organized in terms of indentation and comments
- **75%** There has been only modest innovation attempted
- **50%** The game requires substantial troubleshooting to run, or offers very little innovation
- **25%** The game is almost entirely nonfunctional or offers almost no innovation on top of the basic example
- **0%** No work has been submitted



---

## Attendance and Participation

### Assessed at End of Semester

100 pts (10% of Semester Grade)

This grade will be the average of both components listed below. Students will receive a percentage of the point total above, based on the quality of their performance this semester.

#### **Grading Rubric for Attendance**

- **100%** The student attended all classes, and was never significantly late to class without notice
- **95%** The student missed one or two classes but always provided advanced notice
- **90%** The student had one (1) unexcused absence or tardiness, or missed three (3) classes total
- **80%** The student had two (2) unexcused absences or tardinesses, or missed four (4) classes total
- **70%** The student had three (3) unexcused absences or tardinesses, or missed five (5) classes total
- **60%** The student had four (4) unexcused absences or tardinesses, or missed six (6) classes total
- **50%** The student had five (5) unexcused absences or tardinesses, or missed seven (7) classes total
- **40%** The student had six (6) unexcused absences or tardinesses, or missed eight (8) classes total
- **30%** The student had seven (7) unexcused absences or tardinesses, or missed nine (9) classes total
- **20%** The student had eight (8) unexcused absences or tardinesses, or missed 10 classes total
- **10%** The student had nine (9) unexcused absences or tardinesses, or missed 11 classes total
- **0%** The student had 10 or more unexcused absences or tardinesses, or missed 12 classes total

#### **Grading Rubric for Participation**

- **100%** The student was active in class discussions and participated in class activities
- **90%** The student met all requirements but is encouraged to speak more in class
- **80%** The student is encouraged to participate in more class activities
- **60%** The student was often disengaged, spending class time on unrelated activities
- **30%** The student missed too many classes to participate in a meaningful way
- **0%** The student never attended class

---

## Academic Policies

### **Policy on Academic Integrity and Cases of Academic Dishonesty**

This class will follow the policies of Rider University regarding Academic Integrity, as well as the procedures in addressing cases of Academic Dishonesty. The College's policies on such matters can be found in their entirety at: <http://catalog.rider.edu/policies/code-academic-integrity/>. Academic Dishonesty refers to a misrepresentation of the source or permissions related to the submission of written and creative works. In the area of digital design, this includes visual work, audio work, and written work. If in doubt, feel free to discuss sources with me before submission of work.

### **Academic Success Center**

Students needing supplemental help beyond the scope of this class are encouraged to contact the [Academic Success Center](#) in Suite 237 of the Bart Luedeke Center. The center's services include a writing studio, success coaching and the potential for tutoring services. Email [academicsuccesscenter@rider.edu](mailto:academicsuccesscenter@rider.edu).

### **Academic Disability Policy**

Rider University is committed to providing reasonable accommodations for all students with disabilities. If you are seeking classroom accommodations under the Americans with Disabilities Act or Section 504 of the Rehabilitation Act of Sample Syllabus Statements-Disabilities April 2021 1973, you are required to register with [Student Accessibility and Support Services](#) office (SASS) at the Bart Luedeke Center, Suite 201. SASS can be contacted by email at [accessibility@rider.edu](mailto:accessibility@rider.edu) or by phone at 609-895-5492, To receive academic accommodations for this class, please obtain the proper accommodation form from SASS and meet with me at the beginning of the semester to discuss your accommodations.

### **Class Absence Notice**

It is the student's responsibility to inform instructors of the nature and extent of an actual or anticipated absence. If that is impossible, or if the absence is or will be more than three (3) consecutive class sessions (seven (7) calendar days), the student should contact the Office of the Dean of Students at [deanofstudents@rider.edu](mailto:deanofstudents@rider.edu) or 609-896-5101. Then the dean's office will notify the appropriate faculty member. More information about the procedure for notifying the dean's office of absences can be found at <https://www.rider.edu/about/offices-services/student-affairs/dean-of-students/info-for-students/class-absence-notice>

### **Incomplete Grades**

Students who, as a result of extenuating circumstances, are unable to complete the required work of a course within the term, may request an extension of time from a faculty member. Such extensions of time can be granted only in cases in which illness or another serious emergency has prevented the student from completing the course requirements or from taking a final examination. The request for extension of time must be made prior to the last scheduled class meeting, except in those unusual situations in which prior notification is impossible.

The faculty member shall determine whether to grant the request for a time extension and the type of verification (if any) required to support the request. The faculty member shall specify the time, up to four weeks from the last day of the term, as specified in the academic calendar, by which work must be completed by the student. If the faculty member does agree to the request, the notation "I" (Incomplete) shall be submitted on the grade roll. In those situations where the faculty member has not received a request for an extension of time, the notation "I" (Incomplete) may be submitted on the grade roll by the faculty member when, in his or her judgment, such a determination appears justified. Upon submission of completed required work the faculty member shall submit a Change of Grade form to the Registrar.

Students who, as a result of extenuating circumstances, are unable to submit the completed required work at the end of the four-week period may request an extension of the incomplete grade. The request for an extension of the incomplete must be made prior to the expiration of the four-week period. If the faculty member agrees to the request for an extension of the incomplete, the faculty member shall specify the time, up to a maximum of two weeks from the date of expiration of the four-week period (i.e., six weeks from the last day of the term) by which work must be completed by the student and shall submit an Extension of Incomplete form to the Registrar.

Upon submission of completed required work, the faculty member shall submit a Change of Grade form to the Registrar and assign the course grade. Failure of the Registrar to receive from the faculty member a Change of Grade form or an Extension of Incomplete form at the end of the four-week period, or a Change of Grade form at the end of the six-week period shall result in the automatic assignment of the grade "F," "Z," or "U" by the Registrar.

Students who receive an incomplete in a course that is part of a course sequence must obtain permission from the department chairperson to remain enrolled in the next course in the sequence or they will be removed from that next course.

More information on grade reports can be found here:

<http://catalog.rider.edu/policies/undergraduate/grades/>

### **Courses — Adding, Dropping, Withdrawing, Auditing, Repeating**

Students may add courses through the first week of the regular semester at their own discretion provided the course is still open for registration. Students may drop courses through the second week of the regular semester at their own discretion. In such cases, the courses are deleted from the student's record. After the second week of the semester, a withdrawal from the course is necessary and a 'W' is recorded on the transcript.

Students may withdraw from courses and receive a grade of 'W' during the third through tenth weeks of the semester. The student's academic advisor and financial aid counselor will be notified of class withdrawals by email.

View the course drop/withdrawal policy:

<https://www.rider.edu/tuition-aid/financial-aid/payment-billing/drop-withdrawal-policy>

Spring 2025 withdrawal dates can be found here:

<http://catalog.rider.edu/academic-calendar/#Spring%20semester>

More information on course processes can be found here:

<http://catalog.rider.edu/policies/undergraduate/courses-add-drop/>

Registrar forms can be found here:

<https://www.rider.edu/academics/academic-support/registrar/forms>

---



*Please note that this course syllabus is licensed under a [Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License](https://creativecommons.org/licenses/by-nc-nd/4.0/). This syllabus may not be used for commercial purposes or be distributed as part of any derivative works without my (Matthew DiMatteo's) written permission.*