1. **Learning Objectives:**

   In this unit, students will create a simple but complete version of Pac-Man game while expanding knowledge of the AgentSheets software program. Students will apply basic and advanced design process to identify objects “agents” and interactions “operations”. Throughout this unit, students will be introduced to computational thinking patterns and skills, including basic object interaction, creating object instances, rule based programming, and message sending. Over the course of the unit, the difficulty of the game with ghosts with random movement will be contrasted to the difficulty of the game with ghosts with artificial intelligence.

   In this lesson, students will be finishing and publishing their Pac-Man game.

2. **Standards:**

   ISTE (International Society for Technology in Education) NETS (National Educational Technology Standards)
   - #1a apply existing knowledge to generate new products
   - #1b create original works as a means of personal or group expression.
   - #4b plan and manage activities to develop a solution or complete a project.
   - #4d use multiple processes and diverse perspectives to explore alternative solutions.
   - #6c troubleshoot systems and applications.
   - #2a interact, collaborate, and publish with peers, experts, or others employing a variety of digital environments and media.
   - #2d contribute to project teams to produce original works

   ISTE NETS are referred to by CDE Performance Standards for Teachers #7- Technology, which states, “The teacher will have demonstrated the ability to instruct students in basic technology skills. He/She will: … instruct students in basic technology skills by imbedding them in their standards-based, content instruction (7.5.3)”

   Please check with your district’s technology department to see if there are additional standards at the district or school level.

3. **Anticipatory Set: 5 minutes**

   Play the video clip as students enter the room and get settled. The clip shows a man winning Ms. Pac-Man blindfolded.

   ![Ms_Pacman_Blindfolded_featured_hollywood_blockbuster_video](Ms_Pacman_Blindfolded_featured_hollywood_blockbuster_video)
4. Teaching/Discussion: 15 minutes
Summary of project and AgentSheets

Discussion - Have students compare the “winability” of the students’ versions of Pac-Man with intelligent ghosts compared with their previous versions where the ghosts were just randomly moving. Set this in the history of the game (Historical Note: Pac-Man celebrated his 30th birthday on May 22, 2010!) – originally Pac-Man’s ghosts just followed a certain pattern: [http://www.youtube.com/watch?v=I33NJoq-Qks](http://www.youtube.com/watch?v=I33NJoq-Qks)
People could win this game blindfolded by completing a memorized set of moves for Pac-Man.

Ms. Pac-Man made this much more challenging as her game came with ghosts with artificial intelligence. Though some still could win this one blindfolded too - as you saw on the video which was playing when you came in! (How many times did he have to shoot this video?)

Discuss ways to make improve their games winnable:

To make a winnable game:
- Add counter agent to track number of pellets eaten,
- Add lives – begin with three and game over when third Pac-Man is deflated, and
- Add power pellets and related behavior – reverse Hill Climbing Algorithm.
- Or add doorway to escape maze

To make the game more arcade authentic:
- Make better looking walls – uses “run duplication” feature of AgentSheets
- Add more advanced levels (the original Pac-Man had over 250 levels!)
- Add special limited time pellets (for example, fruit) which are worth more points when gobbled

5. Guided and Independent Practice: 30 minutes
Allow students class time to implement design changes. If time allows, students could be required to make the game winnable, or to make a set number of design changes.

Remind students to save the worksheet periodically and check with the reset button to verify if changes are saved.

**IMPORTANT: Saving the Worksheet**
Students should also be play testing their games as they make modifications. Check understanding of each student.

6. Closure: 15 minutes
Review the computational thinking patterns students used in programming Pac-Man:
- **Collision**: Pac-Man collides with ghosts.
- Artificial Intelligence using **Collaborative Diffusion**
- the *Hill Climbing* algorithm using *Heat values*

Discuss Rubric for grading. Distribute copy of rubric to students if not already done. Review how student’s projects will be graded.

7. **Extension/ Remediation**  
After uploading, have students go to the *Scalable Game Design Arcade* to play each other’s Pac-Man games. You could even include a peer review grade as part of each student’s final grade for the project.