Sample Sokoban Lesson Plans  
Day 1 – 50 minutes  
Scalable Game Design Summer Institute 2011

Note: Red links take you to portions of the tutorial. Blue links provide background information.

1. Learning Objectives:  
In this unit, students will create a simple version of Sokoban game while learning AgentSheets software program. Students will apply design process to identify objects “agents” and interactions “operations”. Throughout this unit, students will be introduced to basic computational thinking, including basic object interaction, stacks, creating object instances, rule based programming, and message sending.

In this lesson, students will create a new AgentSheets project and worksheet, a sokoban, floor and wall agents and will be able to move the sokoban agent in four directions.

2. Standards:  
ISTE (International Society for Technology in Education) NETS (National Educational Technology Standards)  
- #1a apply existing knowledge to generate new products  
- #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.

ISTE NETS are referred to by CDE Performance Standards for Teachers #7- Technology Please check with your district’s technology department to see if there are additional standards at the district or school level.

3. Anticipatory Set / Modeling: 5 minutes  
Quickly demonstrate Sokoban – state the objective of the game and allowed moves. Select one student to play Sokoban while it is being projected on overhead for others to see.

4. Teaching: 10 minutes  
Input – Project Overview and AgentSheets  
Discuss Rubric for grading. Distribute copy to students (optional). Explain how student’s projects will be graded at the end of the week.

Describe components of AgentSheets:  
- Gallery- where agents are  
- Worksheet – where game is created  
- Behavior – how to tell each agent what to do  
And the computational thinking pattern in Sokoban:  
- Push: Sokoban pushes the crates to the goals
Have students decide what the nouns (the agents) and the verbs (the operations) in Sokoban. We will begin by creating the most prominent nouns today – the sokoban, the floor, and the walls, and programming sokoban’s movements.

5. **Guided Practice / Monitoring:** 30 minutes

Demonstrate how to open AgentSheets program and have students make a new worksheet. [Creating A New Project](#)
Check for understanding.

Demonstrate how to create a new agent and have students make 3 new agents: the sokoban, the floor and the wall.

**Creating Agents**
1. [Creating the Sokoban](#)
2. [Creating the Floor](#)
3. [Creating the Wall](#)
Check for understanding.

Demonstrate how to create a new worksheet and have students open a new worksheet. [Creating a Worksheet](#)

Then students create the setting for the game by placing walls, floors (using the rectangle tool) and the sokoban (using the pencil tool):

- [Placing Walls](#)
- [Placing Floors](#)
- [Placing the Sokoban](#)

Students should save the worksheet and check with the reset button to see if it worked.
See **IMPORTANT: Saving the Worksheet** in tutorial

Then have students run the program to see what happens (nothing yet!)
See **Play Test: Running the Program** in tutorial
Check understanding of each student.

Beginning with the up movement, demonstrate how to edit the behavior of the sokoban agent.

**Programming the Sokoban Up Movement**

Ask students to make sokoban agent movements correspond to the arrow keys.

**Moving in the Other Directions**

Ask students to re-run the program to see if movement works.
See **Play Test: Testing the Four Directions of Sokoban Movement** in tutorial.
Check understanding of each student.
6. **Closure: 5 minutes**
   Restate the scope of the project. Tomorrow we will be refining the sokoban’s movements so he cannot walk through walls. We will also be creating the crate agent and learning the computational thinking pattern *Push*.

7. **Extension/Remediation** – students can edit their agents at any time. Encourage students to spend a short time on the initial creation and edit later as desired. **Optional activity:** Have students work individually using various games uploaded to the *Scalable Game Design Arcade* to get an understanding of components of AgentSheets and the types of computational thinking patterns