Topic	**CAPSTONE CLASS: WORLD'S HARDEST GAME**	
Class Description	**Students use their knowledge of sprites, functions, loops, and sound to create the World's Hardest Game.**	
Class	**C8**	
Class time	**45 mins**	
Goal	• Build "The World's Hardest Game" • Review the concepts from the previous classes	

Resources Required	• Teacher Resources
	  ○ code.org login
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen
	• Student Resources
	  ○ code.org login
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen	

Class structure	**Warm Up - Slide show option** **Teacher-Led Activity** **Student-Led Activity** **Wrap Up - Slide show option**	**15 Mins** **8 Mins** **30 Mins** **5 Mins**

WARM UP SESSION - 15mins

Teacher starts slideshow  **from slides 1 to 15**

Refer to speaker notes and follow the instructions on each slide.

Activity details	Solution/Guidelines

Hey <student name>. How are you? Nice to see you! Let's learn something new today, but before we start, do you remember what we are going to learn today? **Run the presentation from slide 1 to slide 10.** **Following are the warm up session deliverables:** • Explain the importance of a capstone class. • Recall last class learnings. • Help the student recall concepts covered so far to design a complete game.	**ESR**: Hi, thanks, yes! *Student recalls from the last class what the teacher mentioned regarding what will be covered in the upcoming session.* Click on the slide show tab and present the slides.
QnA Session	
Question	**Answer**

Identify the right output for the following snippet
 	
for(var i=20;i<400;i=I+50){
var box = createSprite(i,i,20,20)
}	

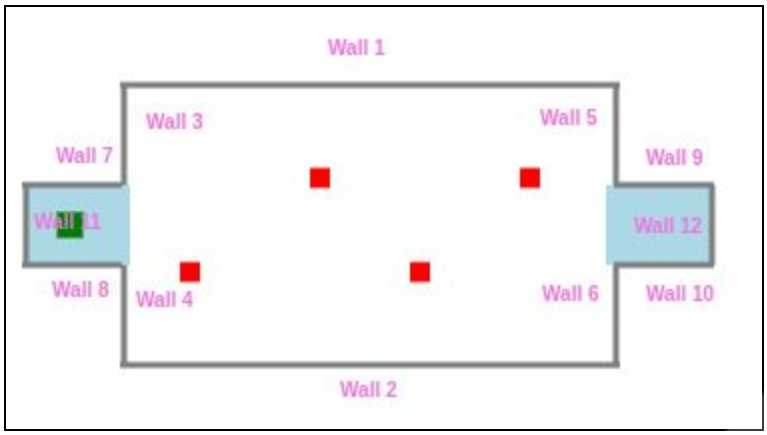
Continue the warm up session	
Activity details	**Solution/Guidelines**
Run the presentation from slide 11 to slide 15 to set the problem statement. **Following are the warm up session deliverables:** • Use tips for teaching students how to approach any complex problem. • Explain the World's Hardest Game that needs to be designed in the class today.	Narrate the story by using hand gestures and voice modulation methods to bring in more interest in students.

<table>
<tr><td colspan="3">Teacher ends slideshow  </td></tr>
<tr><td colspan="3">TEACHER-LED ACTIVITY - 8mins</td></tr>
<tr><td colspan="3">Teacher Initiates Screen Share</td></tr>
<tr><td colspan="3"><u>CHALLENGE</u>
● Decompose “The World’s Hardest Game”
● Ask the student to recall concepts which can be used to build the game</td></tr>
<tr><td colspan="2">Activity details</td><td>Solution/Guidelines</td></tr>
<tr><td>Step 2:
Teacher-led Activity
(15 min)</td><td>Teacher opens the game to talk about it <u>Teacher Activity 1</u>
Did you like playing the game?
What would make the game even harder?</td><td>The student thinks about it.
ESR:
- More red squares
- Higher speed of the red squares</td></tr>
<tr><td></td><td>In today's class, we will be building this game while reviewing all the concepts that we learned in the previous class.
By the completion of this class, hopefully you will have “The World’s Hardest Game” which you can challenge your friends to play. How do you feel about this?</td><td>ESR:
varied</td></tr>
<tr><td></td><td>If you remember, in the Pong Game, we decomposed or broke down the entire game into smaller components. Moving paddle, Bouncing ball, center line, Computer paddle following the ball, scoring system etc.

This is called decomposition. Before working on any project, a programmer breaks down the complex task into</td><td>ESR:
- Walls of the tunnel in which the game is being played.
- Red squares bouncing on the walls
- Green square which could be controlled by the right and left arrow key</td></tr>
</table>

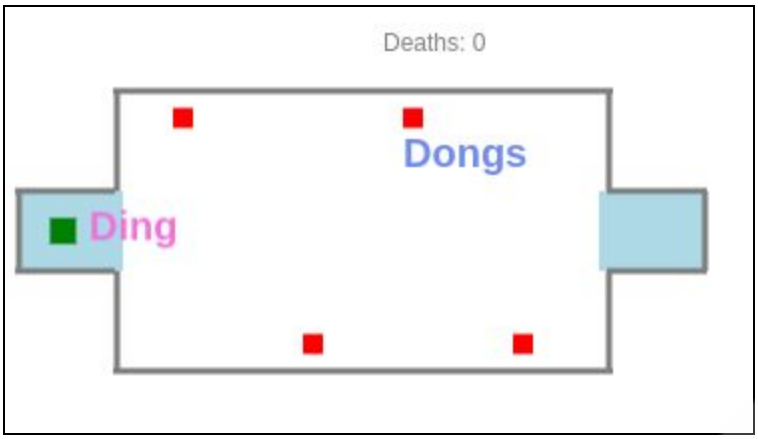
	smaller and simpler tasks. Each task should be simple enough to think on how to approach it. How would you break this game into simpler components?	- Reset game when square touches the green squares - Counting the deaths - Adding sounds
	Awesome. Optionally, we can also add more levels to the game by increasing the challenge for the user. How would we add more levels?	**ESR:** Using Game States
	Let's start coding to program for each small component of the game. You will observe how the entire game gets built by solving small components in the game. Let's get started.	-
Teacher starts slideshow :Slide 16-17 (Only 1 slide for this Activity)		
Run the presentation slide 16 to set the student activity context.		
Here's a challenge for you, you need to design the world's toughest game. Keep in mind all the concepts you have learned so far regarding sprite objects, properties, functions, conditional statements, game states etc. Can you tell me what are the various steps involved and how would you start?		**ESR:** We will start by - creating 12 walls - 1 ding - 4 dongs - assign velocityY to dongs - use if condition to move the ding sprite - use game state - add sound effects to the game - score the player

Awesome! Go ahead and share your screen, let's get started!		*The student shares his/her screen.* *Opens the Student Activity, remixes the code and adds more code to it.*
Teacher ends slideshow 		
Teacher Stops Screen Share		
STUDENT-LED ACTIVITY - 8 mins		
• **Ask Student to press ESC key to come back to panel** • **Guide Student to start Screen Share** • **Teacher gets into Fullscreen**		
ACTIVITY • **Student code to build the complete game** • **Add optional challenges for the student**		
Step 3: Student-Led Activity (15 min)	*Do you remember how sprite is created?* Let's use **createSprite()** function to create all the walls in the game. *Help the student create the walls of the tunnel using sprites.*	*Student opens Student Activity 2* *Using createSprite() function.* *The student writes code to create the walls of the tunnel in the game and position them on the game.*




```
1  var wall1 = createSprite(190,120,250,3);
2  var wall2 = createSprite(190,260,250,3);
3  var wall3 = createSprite(67,145,3,50);
4  var wall4 = createSprite(67,235,3,50);
5  var wall5 = createSprite(313,145,3,50);
6  var wall6 = createSprite(313,235,3,50);
7  var wall7 = createSprite(41,170,50,3);
8  var wall8 = createSprite(41,210,50,3);
9  var wall9 = createSprite(337,210,50,3);
10 var wall10 = createSprite(337,170,50,3);
11 var wall11 = createSprite(18,190,3,40);
12 var wall12 = createSprite(361,190,3,40);
13
```

	*Help the student create the **green square** which can be controlled using right and left arrow keys.* *Recall conditional programming and key events for the student.*	*The student writes code to create a red square sprite and controls it using right and left arrow keys.*



Create Green Square

```
14    var ding = createSprite(40,190,13,13);
15    ding.shapeColor = "green";
16
```

Control Green Square

```
54    if(keyDown("right")){
55      ding.x = ding.x + 2;
56    }
57    if(keyDown("left")){
58      ding.x = ding.x - 2;
59    }
60
```

	You are doing awesome so far. *Get the student to create the red squares which bounce off the top and bottom walls.*	*Student creates the four green square which bounce off the top and bottom walls*

```
var dong1 = createSprite(100,130,10,10);
dong1.shapeColor = "red";
var dong2 = createSprite(215,130,10,10);
dong2.shapeColor = "red";
var dong3 = createSprite(165,250,10,10);
dong3.shapeColor = "red";
var dong4 = createSprite(270,250,10,10);
dong4.shapeColor = "red";

dong1.velocityY = 8;
dong2.velocityY = 8;
dong3.velocityY = -8;
dong4.velocityY = -8;
```

```
dong1.bounceOff(wall1);
dong1.bounceOff(wall2);
dong2.bounceOff(wall1);
dong2.bounceOff(wall2);
dong3.bounceOff(wall1);
dong3.bounceOff(wall2);
dong4.bounceOff(wall1);
dong4.bounceOff(wall2);
```

	Help the student add logic such that the red square resets its position when it touches any of the green squares.	*The student writes code to reset the red ball's position when it touches any of the green squares.*

```
if(ding.isTouching(wall11)||
   ding.isTouching(wall12)||
   ding.isTouching(dong1)||
   ding.isTouching(dong2)||
   ding.isTouching(dong3)||
   ding.isTouching(dong4))
{
   ding.x = 40;
   ding.y = 190;
   count = count + 1;
}
```

	Guide the student to create a scoring system where deaths are counted.	*The student creates a count variable and increases it after every death.* *The student displays it as deaths.*

```
30
31   var count = 0;
32
```

```
61   if(ding.isTouching(wall11)||
62      ding.isTouching(wall12)||
63      ding.isTouching(dong1)||
64      ding.isTouching(dong2)||
65      ding.isTouching(dong3)||
66      ding.isTouching(dong4))
67   {
68      ding.x = 40;
69      ding.y = 190;
70      count = count + 1;
71   }
```

	Help the student choose a sound, load it, and then play it in a loop.	*Student uses resource from* ***Student Activity 2*** *Student chooses a sound, uploads it and then plays it in a loop.*

```
33   playSound("sound123.mp3", true);
34
```

	(Optional) You can add levels to your game using gameState. *Increase the velocity of the moving squares at higher levels.*	*Student codes to create higher levels in the game.*

Teacher Guides Student to Stop Screen Share

Quiz time - Click on in-class quiz	
Question	**Answer**
To make the red squares (dong) move vertically alternatively, we should: A. Give each of them opposite velocityX values B. Give each of them opposite velocityY values C. Give each of them opposite x values D. Give each of them opposite y values	**B.**

Identify the correct syntax to move the ding to the right side on press of the right key
A. if(keyDown("right")){
ding.x = ding.x + 2;
}
B. if(keyDown("right")){
ding.x = ding.y +2;
}
C. if(keyDown("right")){
ding.y = ding.y + 2;
}
D. if(keyDown("right")){
ding.x = ding.y - 2;
}
	A.

End the quiz panel	
WRAP UP SESSION - 5 Mins	
Teacher starts slideshow  **from slide 18 to slide 28**	
Activity details	**Solution/Guidelines**

Run the presentation from slide 18 to slide 28. **Following are the warm up session deliverables:** • **Explain the facts and trivias** • **Next class challenge** • **Project for the day** • **Additional Activity**		Guide the student to develop the project and share with us.
Teacher ends slideshow 		
FEEDBACK • **Appreciate the student for their efforts** • **Identify 2 strengths and 1 area of progress for the student**		
Step 4: **Wrap-Up** **(5 min)**	So, in this project class we reviewed the concepts we have learned so far. How was your experience? You can now add higher levels of challenges to the game and invite your friends to play them. Congratulations! You are a game developer now!	**ESR:** varied
	Congratulations! You have reached a milestone. Are you ready to face one of the toughest challenges till date? To successfully complete this challenge, you have to apply the programming constructs learnt during the past few classes and create the World's Hardest game. "	-

Project Overview	**WORLD'S HARDEST GAME** **Goal of the Project:** In Class 8, you reviewed the concepts from the previous classes and started to build the T-rex game. In this project, you will be designing your own game! **Story:** Every game has a few simple components… The player playing the game should have a goal. It should give you obstacles to overcome. And it should offer you feedback on how you are doing in the game. Additionally, good games also have some story behind them. With all these points in mind, design your own game with your own character, goals, obstacles, feedback and a story. I am very excited to see your project solution and I know you will do really well. Bye Bye!	*Students engage with the teacher over the project.* We expect the student to apply their creativity in the project.
Teacher Clicks	✖ End Class	
ADDITIONAL ACTIVITIES		

- **Ask Student to press ESC key to come back to the panel**
- **Guide Student to start Screen Share**
- **Teacher gets into Fullscreen**

Activity details		Solution/Guidelines
Additional Activity	*Encourage the student to write reflection notes in their reflection journal using markdown.* Use these as guiding questions: ● What happened today? - Describe what happened - Code I wrote ● How did I feel after the class? ● What have I learned about programming and developing games? ● What aspects of the class helped me? ● What did I find difficult?	*Student uses the markdown editor to write her/his reflection as a reflection journal.*

Activity	Activity Name	Links
Student Activity 1	Game	https://studio.code.org/projects/gamelab/bcclZAj_KEpCFwt_m9I5cWZySsIHHkBk13s_I0ypFXw

Student Activity 2	Project Class Activity	https://studio.code.org/projects/gamelab/izZqcra9kD5RTYWPeqxcp-IGQ_Mp-Gdyk9Z8KLABf08/edit
Student Activity 3	Sound Resource	https://freesound.org/people/djgriffin/sounds/251284/
Teacher Activity 1	Game	https://studio.code.org/projects/gamelab/bcclZAj_KEpCFwt_m9I5cWZySsIHHkBk13s_I0ypFXw
Teacher Activity 2	Reference Link	https://studio.code.org/projects/gamelab/bcclZAj_KEpCFwt_m9I5cWZySsIHHkBk13s_I0ypFXw/edit
Teacher Reference visual aid link	Visual aid link	https://curriculum.whitehatjr.com/Visual+Project+Asset/PRO_VD/c8withclue.html
Teacher Reference In-class quiz	In-class quiz	https://curriculum.whitehatjr.com/Visual+Project+Asset/PRO_VD/Pro-C8_Deepali.docx.pdf