Game Overview
2015 Middle School Program
V1.1
Goals

• Part 1: Game Overview
  – To be used at the end of Week 1
  – Introduces the game in preparation for the “Acting out the Game” activity during Field Day
  – Additional details are available in the game manual

• Part 2: Game Specific Functions Tutorial
  – To be used at the end of Week 2
  – Provides practice exercises for using some of the Game Specific functions
Your Mission

• Conducting Optical Research on Nearby Asteroids (Corona)
  – Your Mission: Use a robotic satellite to take pictures of “points of interest” on an asteroid
  – Your Goal: Collect and upload as many new pictures as possible while avoiding effects of solar flares
Your Tasks

• Command your robotic SPHERES to take pictures of Points of Interest (POIs) on an asteroid
Your Tasks, continued

- Create a plan to protect your SPHERES, your pictures, and your points during solar flares (more details later)

In this example:
- Red SPHERES moved to shadow zone
- Blue SPHERES powered off

Yellow flash represents solar flare
Game Components/Layout

- Asteroid
  - with Points of Interest
- SPHERES
  - SPHERES can only move in the x,y plane
  - SPHERES start with 6 points
- Memory packs
- Zones
  - Danger Zone
  - Inner Picture Taking Zone
  - Outer Picture Taking Zone
  - Shadow Zone
    - Opposite side of the asteroid from the sun

<table>
<thead>
<tr>
<th>SPHERES Satellite Starting Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>X [m]</td>
</tr>
<tr>
<td>Y [m]</td>
</tr>
<tr>
<td>Z[m]</td>
</tr>
</tbody>
</table>
Points of Interest (POI) locations on the asteroid

- 3 sets of Points of Interest (POIs)
- POIs change every 60 seconds as indicated by color changes
- Locations of each set of 3 POIs are the same.

<table>
<thead>
<tr>
<th>POI number (#)</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(0.0, 0.2, 0.0)</td>
</tr>
<tr>
<td>1</td>
<td>(0.0, -0.2, 0.0)</td>
</tr>
<tr>
<td>2</td>
<td>(-0.2, 0.0, 0.0)</td>
</tr>
</tbody>
</table>
Initial Conditions and End of Game

• Initial Conditions
  – Each SPHERES starts with 6 points
  – Game time: 180 seconds
    • Three 60 second periods
  – Virtual fuel: 100 % (60 seconds)
    • 60 seconds thruster firing time

• End of the Game occurs when game is out of time after 180 seconds

• Final score is based on:
  – Pictures you take and upload
  – Minus the damage obtained from solar flares and collisions
Taking Pictures – SPHERES Location and Orientation

• To take a picture the SPHERE must be:
  – In the Inner Zone or Outer zone
  – Facing the POI with acceptable tolerances★
  – Meet criteria for acceptable alignment with POI★
  – Call game function: “takePic” (described later)

★See game manual

Represents the pointing direction (side with Camera)
Taking Pictures – Picture Type Restrictions

• In each 60 second period
  – You may take only one picture of each POI from each zone (maximum two pictures of each POI)
  – There are 6 possible pictures
    • 3 POIs
    • 2 pictures of each POI

Example:
Two pictures of green POI # 0 maximum
One picture from each zone

[Diagram showing inner and outer zone pictures]
Taking Pictures - Memory slots

• Memory must not be full
  – Start with 2 memory slots
    • only 2 photos can be stored at a time
  – Can add memory slots by picking up Memory packs (described later)
  – If memory is full camera is disabled until photos are uploaded
Uploading Pictures

- Conditions to upload a picture back to earth
  - SPHERES must be Outside the picture taking zones or in the Shadow Zone
  - Call game function “uploadPic”

SPHERES Must be outside the picture taking zones to upload a picture
After Taking and Uploading Pictures

• After taking a picture
  – Camera is disabled for 3 seconds
  – Cannot take another picture for 3 seconds

• After starting to upload a picture
  – Camera is disabled for 3 seconds
  – Cannot take another picture for 3 seconds
### Points awarded for pictures

<table>
<thead>
<tr>
<th>Description</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>A picture is attempted (The function takePic is called, the camera is not disabled, but picture taking criteria are not met)</td>
<td>0.01 points</td>
</tr>
<tr>
<td>Picture meets criteria and is stored in memory (not yet uploaded)</td>
<td>0.1 points</td>
</tr>
<tr>
<td>A picture is uploaded to Earth</td>
<td>2.0 points</td>
</tr>
<tr>
<td>Inner Zone picture:</td>
<td>3.0 points</td>
</tr>
<tr>
<td>Discover Bonus (For uploading first picture of a POI taken from inner or outer zone)</td>
<td>0.5 points</td>
</tr>
</tbody>
</table>

Example: Take and upload a valid picture in the inner zone and get the discover bonus:

\[
0.01 + 0.1 + 2.0 + 0.5 = 2.61 \text{ points}
\]

(Picture taken) (Picture is good and is stored) (Picture is uploaded) (Discover Bonus)
Solar Flares

- Number of Solar flares per game: 2
- Solar Flare duration: 3 seconds
- When:
  - Flares occur at random times during each game
  - No solar flares in first 30 seconds
  - Flares are at least 30 seconds apart
• Players can receive a 30 second warning before a solar flare arrives.

• This warning can be received by calling the function: “getNextFlare” (described later)
  – Calling this function during the 30 second period prior to the next solar flare:
    • Gives the number of seconds until the next solar flare
  – Calling this function any other time:
    • Will return -1
Powering SPHERES off/on

• Powering off the satellite outside the shadow zone during a solar flare reduces damage to a SPHERES (and saves points)

• When the SPHERES is powered off
  – Must wait 5 seconds for their instruments to turn on and warm up
  – Cannot take pictures
  – Cannot stop their satellites from drifting (collisions may occur)

• To power off:
  – Game function “turnOff” must be called

• To power back on:
  – Game function “turnOn” must be called
Solar Flare penalties

- During a Solar flare if SPHERES are not protected by the asteroid shadow (if they are outside shadow zone) they will:
  1. **Lose Pictures:** All pictures that have not been uploaded are lost
  2. **Lose Points:**
     - Less damage to the SPHERES *if the SPHERE powers down* before the solar flare hits (and so fewer points lost)

<table>
<thead>
<tr>
<th>SPHERE location/condition during Solar Flare</th>
<th>Points (deducted per second)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPHERE- outside shadow zone / powered</td>
<td>-1.0</td>
</tr>
<tr>
<td>SPHERE- outside shadow zone / un-powered</td>
<td>-0.5</td>
</tr>
<tr>
<td>SPHERE- inside shadow zone</td>
<td>0</td>
</tr>
</tbody>
</table>
Memory Packs

- Memory packs are identical and have 1 memory slot each.
  - Each slot = ability to store 1 more photo
- Memory packs may be picked up by either SPHERE
  - Once picked up by one team it is no longer available for pick up by the other team.
- The locations of the memory packs are shown below
- You may choose to pick up 1 or 2 memory packs or you may choose not to pick up memory packs

<table>
<thead>
<tr>
<th>Memory pack number (#)</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(-0.5, -0.6, 0.0)</td>
</tr>
<tr>
<td>1</td>
<td>(-0.5, 0.6, 0.0)</td>
</tr>
</tbody>
</table>
Picking up Memory Packs

• Locate the memory pack
• Stop the SPHERES
  – within 0.05 m of the location of the pack
• Rotate SPHERES 90°
  – to pick up the memory pack
• Use game-specific function Player Me/Other has pack # (described later)
  – to check whether the memory pack has been picked up
Satellite location and Zone Boundaries

- Satellite location is always the center of the Satellite
- Example: Shadow Zone
  - SPHERES center must be inside the shadow zone
  - It is okay for part of the satellite to be outside the shadow

Both satellites are considered inside the shadow zone
Zone Boundary Details, continued

• Collision with Asteroid
  – If the center point of your satellite enters the Danger Zone, it is considered a collision with the asteroid.

• Penalties
  – Fuel penalty per second of collision
  – subtract 0.1 points per second per POI contacted
Collision avoidance

• If two SPHERES are moving to the same position in the game arena, the SPHERES controllers will take over to initiate motion to prevent a collision between the satellites

• SPHERES may be prevented from successfully completing preprogrammed tasks such as moving into the shadow zone, in order to avoid a collision
  – Keep this in mind when selecting your target position in the shadow zone
Fuel Use, Fuel Penalties

- Fuel is used when thrusters are fired:
  - Motion commanded by player
  - SPHERES motion during:
    - Out of bounds
      - Out of game arena
      - Inside danger zone
    Note: Amount used depends on speed of SPHERES at time of event
  - Additional Out-of-bounds penalty
    - Fuel penalty = 2.5 * (seconds out of bounds)
Zone Boundary Exploration

• Shadow Zone

<table>
<thead>
<tr>
<th>Shadow Zone</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>X [m]</td>
<td>0.0; +0.64</td>
</tr>
<tr>
<td>Y [m]</td>
<td>-0.2; +0.2</td>
</tr>
<tr>
<td>Z[m]</td>
<td>0.0</td>
</tr>
</tbody>
</table>

• Circular Zones

<table>
<thead>
<tr>
<th>Circular Zone Boundaries (center=0,0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Danger Zone [m]</td>
</tr>
<tr>
<td>Inner Zone [m]</td>
</tr>
<tr>
<td>Outer Zone</td>
</tr>
</tbody>
</table>

- Danger zone example:
  - outer radius = 0.31

• Answer the following questions
  - Is the ✗ in the inner or outer zone?
  - What coordinates could you use to move your SPHERE near the ✗?
  - (repeat questions above for the ▼)
  - What coordinates could you use to move your SPHERE in the shadow zone?
Start to plan your strategy

As you plan your strategy, here are some choices to make:

• Pictures
  – Which POI do you want to take a picture of first?
    • What zone do you want to take the picture from?
  – When do you want to upload pictures?

• Avoiding Solar Flares
  – During the first solar flare do you want to…
    • Go to the shadow zone, power off, neither
    • At what number of seconds remaining before the solar flare do you head to the shadow zone or power off?
  – What about during the second solar flare?

• Memory Packs
  – Do you want to pick up 0, 1 or 2 memory packs?
    • When do you want to pick up a memory pack?
    • Which memory pack do you want to pick up?
End of Part 1 materials

This slide has been intentionally left blank.
Part 2: Game Specific Functions

- This tutorial will introduce some of the Game Specific functions for CoronaSPHERES_MS_2D

- Game Specific functions for CoronaSPHERES_MS_2D are split between two tabs as shown here

- It is not necessary to use all of the game specific functions provided to complete game objectives
Part 2: Game Specific Functions
CoronaSPHERES MS- Actions

- Create a new project
  - For “Game” select: CoronaSPHERES_MS_2D
- Click on the CoronaSPHERES MS- Actions tab
  - You will see the three game functions shown below
Part 2: Game Specific Functions

takePic

- The functions: “takePic” and “uploadPic” are introduced using 3 simple examples

- Example 1: Taking nonvalid pictures
  - Drag a “Take Pic: POI 0” block into the loop
  - Simulate and View Results
  - The SPHERE will attempt to take a pictures from its initial position
    - Picture taking criteria are not met here
  - Camera will be turned off for 3 seconds after each attempted picture
  - You will see 0.01 points added to your score for each attempted picture
    - 0.01 points are awarded for attempted pictures that do not meet picture taking requirements
Part 2: Game Specific Functions

takePic

- Example 2: Taking valid pictures
  - Review the CoronaSPHERES game manual for how to take a valid picture
  - Add code to move your SPHERE (use both setPosition target and setAttitude target) to a correct picture taking position for POI #0.
    - You need to figure this out on your own
  - Simulate and View Results
  - When your SPHERE takes a picture **after it has moved to meet picture taking requirements** you will see 0.11 points added to your score
  - Only one valid picture is possible from this one position
    - All other pictures attempted from this position will add invalid picture scores of 0.01 points
Part 2: Game Specific Functions

takePic and uploadPic

• Example 3: Uploading valid pictures
  – Before you can upload a picture you must move to an upload location (review game manual for details)
    • You need to figure this out on your own
  – Add a conditional statement to tell the SPHERE when to move to the upload position
  – Use the uploadPic function to upload a picture
  – Simulate and View Results
  – When your SPHERE successfully uploads a picture points will be added to your score
  – The example shown gained 3.6 points:
    • 3.0 points since the uploaded picture was taken from the outer zone
    • 0.5 points as bonus points
    • 0.01 points for taking an non valid picture
Part 2: Game Specific Functions
CoronaSPHERES MS- Information

• Next click on the CoronaSPHERES MS –Information tab
• The information returned by these game functions gets stored as variables.
  – We will use *getNextFlare* as an example and the others follow a similar pattern
• First – check the Game Functions document (API) to see what type of data is returned by each function.
  – Some are ints and some are floats.
• *int getNextFlare()*
  – means the function *getNextFlare* returns an integer.
Part 2: Game Specific Functions
getNextFlare

- On the init page:
  - Create an int variable, ex. NextFlare, to store the value returned from the function getNextFlare

- On the main page:
  - Assign your variable (ex. NextFlare) equal to the value of the function “getNextFlare”
    - Use “Select = 0“ block
    - Go to the CoronaSPHERES – Information menu to find the function block “getNextFlare”
Part 2: Game Specific Functions
getNextFlare

• The information returned by calling the function `getNextFlare` is stored in the variable called `NextFlare`.

• This information can be used in conditional statements
  – An example of an `if-then` statement is started for you as shown to the right
  – How might you use this in your game code?
Part 2: Game Specific Functions

getNextFlare, continued

• To be sure your code is working correctly and/or to help troubleshoot you can add a debug statement
  – Remember you can use debug statements to print variables as follows. Use the correct symbols for each data type:

<table>
<thead>
<tr>
<th>Data type</th>
<th>symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Float</td>
<td>%f</td>
</tr>
<tr>
<td>Int</td>
<td>%d</td>
</tr>
</tbody>
</table>

  – Use the following format inside the debug block:
     “text text text symbol”, variable

For this example:

  “NextFlare in %d”, NextFlare

(remember the position of “” quotation marks are important!)
Part 2: Game Specific Functions
getNextFlare, continued

- Simulate and view results
- Look in the console
  - This function will return the number of seconds until the next solar flare up to 30 seconds before the flare occurs
  - First you will see the message:
    - SPH1: NextFlare in -1
  - Then you will see
    - SPH1: NextFlare in 30
    - SPH1: NextFlare in 29
    - SPH1: NextFlare in 28, etc
Part 2: A useful function found in the SPHERES Controls Tab: \texttt{getTime} \\

- Check game time using \texttt{getTime} \\
- Note that the function \texttt{getTime} stores information as an \texttt{unsigned int}. \\

- Introduction to \texttt{unsigned ints} \\
  - An unsigned int is a type of integer that cannot be negative \\
  - Unsigned ints include only positive whole numbers or 0 such as: 
    - 0, 1, 2, 3, 4
Part 2: A useful function: getTime, continued

- On the init page
  - Create a variable, ex. gametime, to store the value returned by the function `getTime`
  - **Select type: unsigned int**

- On the main page:
  - Assign your variable (ex. gametime) = the value returned by the function “getTime”
    - Use “Select = 0 “ block
    - Go to the SPHERES Controls menu to find the function block “getTime”
Part 2: Another useful variable: getTime, continued

- Use the variable gametime in conditional statements in your code
  - Such as if gametime < 60 seconds then...
- But for now you can test to see how the function works by adding a debug statement
  - Note that an unsigned int uses a different symbol in the debug statement as shown in the table to the right.

For this example:

```
"gametime=%u", gametime
```

(remember the position of "" quotation marks are important!)

- Simulate and View results- look at what prints in the console

<table>
<thead>
<tr>
<th>Data type</th>
<th>symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Float</td>
<td>%f</td>
</tr>
<tr>
<td>Int</td>
<td>%d</td>
</tr>
<tr>
<td>Unsigned Int</td>
<td>%u</td>
</tr>
</tbody>
</table>
• Congratulations!
  – You know the basics about the game
  – You have some experience with how to use the Game Specific Functions!
<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>Original release</td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>6/1/15</td>
<td>Added takePic examples (slides 30-32) Corrections/changes made to slide 6, 8 (Sphere initial position, Initial score)</td>
</tr>
</tbody>
</table>