CUTTING POINTS

1) Struct needs to be correct at all times. If they are not updating correctly, graphics on screen will lag or not work entirely. (so between tasks and structs - check data is updating correctly with some form of unit test)
2) Collisions with objects - need to make sure collisions and the positions of objects are correct. This also ties in with making sure the structs are updating correctly, so there is some overlap. However, I also need to check that collisions with satchels and the wall end the game, collisions with the satchel and shield destroy the satchel, etc. This can be checked visually, in addition to having some type of unit test to make sure the numbers in my struct are updating correctly.
3) Satchel not firing off when current one is off screen - should be checking for whether or not a satchel is present. If there isn't one, fire another. Can be checked visually like the last one, in addition to checking a boolean or something I will implement as soon as the current satchel is no longer present.

Cutting point 1:
- Move slider for a certain amount of time, check if platform moves correct distance - not sure yet exactly how I am going to check this - Would probably pass
- Press shield button, see if shield instantly pops up (if enough energy) - Would pass

Cutting point 2:
- Move platform into canyon wall, check program acknowledges collision (when I get far enough, this can be replaced with a visual check that the game ends - I have reached this point in my code now and I can set a breakpoint to verify this happens) - Would pass
- Let satchel hit platform, check that program acknowledges collision (also game ends when I get that far) - Would fail

Cutting point 3:
- When there is no satchel on screen, can use debugger to see if program acknowledges this and is going to fire another the next time it enters the data monitor task - Would fail
- When a satchel is on the screen, can check the boolean or whatever I decide to use to make sure the program knows it is there (otherwise it will just start firing off a ton of them) - Would fail

- Shoot at wall, part of wall should break - Would fail
- Railgun should disable after shot until railgun shot is off screen - Would fail
- Shoot while moving, platform should keep moving - would fail
- Let railgun shot hit platform, should destroy platform - would fail
- Tap railgun button, and hold after, first shot should go slow and second shot should go faster - would fail
- When foundation has one "shot" left, left LED should start blinking - would fail
- Hold far left of slider, then middle left, far left should see right LED is brighter and middle left dimmer

- Accelerate slider continuously into either wall, should see "GAME OVER" when slider is going fast enough - would pass
- Hold railgun shot button, press shield button, shield should pop up - would fail
- Hold shield button, shield should only pop up for rising edge - would pass

FUNCTIONAL TESTS:
1) Hold far left of slider, should accelerate to left (when stationary). P
2) Hold far right of slider, should accelerate to right (when stationary). P
3) Hold slider opposite of direction of motion, should decelerate. P
4) Press shield button, shield should instantly pop up. P
5) Hold railgun shot button for at least two seconds, should shoot full power. P
6) Tap railgun shot button, should shoot low power. P
7) Let satchel hit slider, game should end. P
8) Let slider hit wall at high speed, game should end. P
9) Shoot foundation twice, game should end and left LED should come on. P
10) Shield when satchel is falling towards platform, satchel should reset and platform should not be hit. P
11) Shoot foundation once, left LED should blink. F
12) Press shield, right LED should get dimmer. F

WEEKLY SUMMARY

This week I finished most of the rest of my project. I implemented the satchels and their collisions, the railgun, a win condition, and got basically everything display-wise working. The only required parts that I have remaining are the two LEDs.

SUMMARY EFFORT and ESTIMATE NUMBERS
I did approximately 11.5 hours of work this week. I am now at a total of about 24.5 hours out of 29.5 expected hours, about 83% of my expected work. I only have about 5 hours worth of expected work, but I think it will take me less time than that based on the progress I have already made with my LEDs.

IN SCOPE WORK ITEMS

I finished every task except for the unit test plan, LED functionality, and fine tuning. I don't expect these last couple tasks to take me very long, so I am happy with my progress.

TASK	EXPECTED	ACTUAL	EXPECTED TOTAL	ACTUAL TOTAL	COMPLETE?
PROJECT PLANNING	3	2.5	3	2.5	Y
UNIT TEST PLAN	3	1	6	3.5	N
BUTTON FIFO/SLIDER RESPONSE	2	2	8	5.5	Y
IMPLEMENT PHYSICS	8	10	16	15.5	Y
DATA MONITOR TASK	1.5	1	17.5	16.5	Y
DISPLAY TASK (not actualy display)	2	3	19.5	19.5	Y
DISPLAY W/ GRAPHICS	4	4	23.5	23.5	Y
LED TASK and FUNCTIONALITY	3	1	26.5	24.5	N
FINE TUNING (better graphics, messing with different settings to polish final projec	3		29.5		N

RISK REGISTER

This week I added the risk of my satchel collision with the slider not being recognized, as I am unsure if I have covered every case. I tested as many as I could think of.

Slider sampled too slowly	10	80	800	3/24/23		R	Plan ahead to make sure tasks are prioritized well
Incorrect task diagram	100	100	10000	3/24/23	Mitigated	M	Checked with professor or TA
Losing track of time	30	80	2400	3/24/23	Mitigated	M	Going to make notes of what I have to do, write down how much time I have left to finish
Task switching (Do I need monitor task?)	5	60	300	3/24/23		R	Note that you can probably get rid of monitor task if you notice task switching is slowing program down too much
Platform clipping through side boundary	10	80	800	4/7/23	Resolved	R	I believe I covered all edge cases
Satchel collision with slider	10	30	300	4/21/23	Mitigated	M	Pretty sure I set up my conditions correct as I tested them, however not sure all edge cases are covered