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
- Press shield button, see if shield instantly pops up (if enough energy)

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)
- Let satchel hit platform, check that program acknowledges collision (also game ends when I get that far)

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
- 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)

WEEKLY SUMMARY

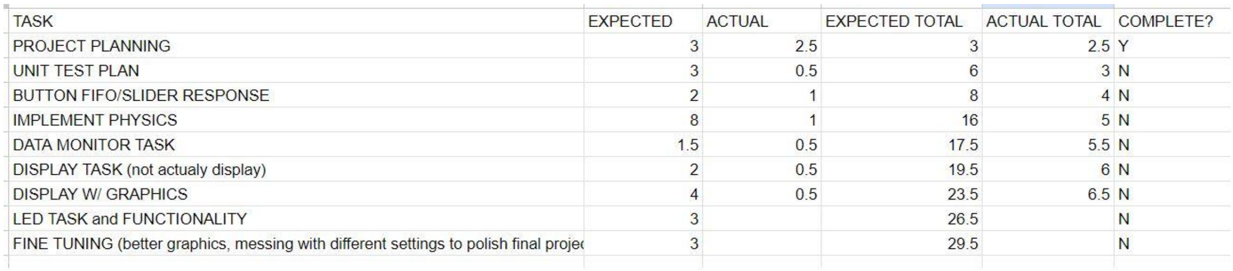
This week, I was able to get my slider to control my platform. I was able to do the physics for the platform, where the slider controls the acceleration applied to the platform. The display appears to accurately show where the platform is based on the physics struct, and the platform accelerates and decelerates based on how I touch the slider. The platform bounces off the sides of the canyon, and if it is going fast enough when it collides it switches to a “game over” screen.

SUMMARY EFFORT and ESTIMATE NUMBERS

This week, I completed about four hours worth of work. In these four hours, I completed approximately six hours of my expected work. I adjusted my expected total to reflect this. These are just estimates, as I completed parts of different tasks without fully completing tasks (see “complete” column of spreadsheet), so it is hard to tell how much of my expected work I actually completed. Based on these estimates, I have completed 6.5/29.5 = 22% of my expected work. If I factored in the roughly 1 hour I spent writing this up, this becomes 7.5/29.5 = 25.4%.

IN SCOPE WORK ITEMS

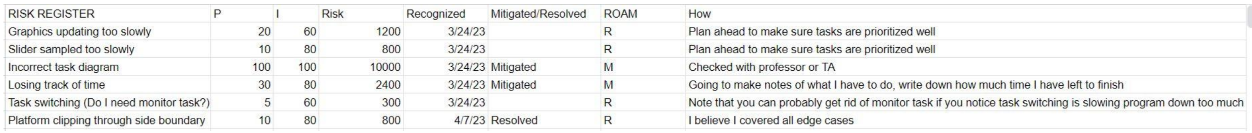
No new tasks were fully completed this week. However, in the “actual” column I updated about how long I spent on each task this week. Getting the slider to work with physics and the display required me to work on multiple different tasks.



TASK	EXPECTED	ACTUAL	EXPECTED TOTAL	ACTUAL TOTAL	COMPLETE?
PROJECT PLANNING	3	2.5	3	2.5	Y
UNIT TEST PLAN	3	0.5	6	3	N
BUTTON FIFO/SLIDER RESPONSE	2	1	8	4	N
IMPLEMENT PHYSICS	8	1	16	5	N
DATA MONITOR TASK	1.5	0.5	17.5	5.5	N
DISPLAY TASK (not actualy display)	2	0.5	19.5	6	N
DISPLAY W/ GRAPHICS	4	0.5	23.5	6.5	N
LED TASK and FUNCTIONALITY	3		26.5		N
FINE TUNING (better graphics, messing with different settings to polish final projec	3		29.5		N

RISK REGISTER

I added a new risk this week, and was also able to resolve it. I realized my platform could clip through the boundary wall if it is moving fast enough, so I had to add a couple edge cases to resolve it.



RISK REGISTER	P	I	Risk	Recognized	Mitigated/Resolved	ROAM	How
Graphics updating too slowly	20	60	1200	3/24/23		R	Plan ahead to make sure tasks are prioritized well
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