Due to the circumstances of our situation the project is going to be smaller (not necessarily easier)

**TASK:** Fix my broken pong game TEST.asm (This is called getting Tom Sawyered)

10% Setup and install the TASM package and DOSBOX

Each team is going to be assigned code for an assembly game

The first goal is to make the game given to you your own.

10% Change the controls for the game

10% Change the Colors of the game

10% Change the display of the game (Make the screen bigger and change resolution draw/resize things on screen)

10% Change the speed of the game

30% Report

20% Add something new to the game. This can be one major thing such as an AI opponent (that is beatable), several powerups or several smaller things such as adding a second ball in pong, selectable difficulty levels, additional players (1v1v1v1? 2v2?).

BONUS-10% Create a better way to refresh the screen. As it stands the method used causes severe flickering. Adjusting the refresh timer is not a guaranteed fix. If you are attempting this state this within the discord server in the appropriate channel.

IF YOU WISH YOU MAY BUILD A GAME FROM THE GROUND UP BUT YOU MUST EXPLAIN THE CODE THAT HAS BEEN GIVEN TO YOU AND EXPLAIN WHY YOU DECIDED TO MAKE A COMPLETE OVERHAUL.

Report Format:

3 main sections:

Explanation of base game (60% of Report Grade)

* Explain each mechanic of the game and how it works
* What interrupts does it use and why?
* How does it use memory?
* Any alternate ways of completing the task?

Explanation of additions (30% of report grade)

* Explain each mechanic added and how it works
* Did the structure of the original code alter your approach to adding your new mechanic?

Reflection on Assembly (10%)

* I know that it has been rough, and I would have liked to give more projects like this over the semester but what have you learned about assembly and have you developed an appreciation of its place in computer programming. Feel free to add any new concepts that you may have learned about while working on the final project.