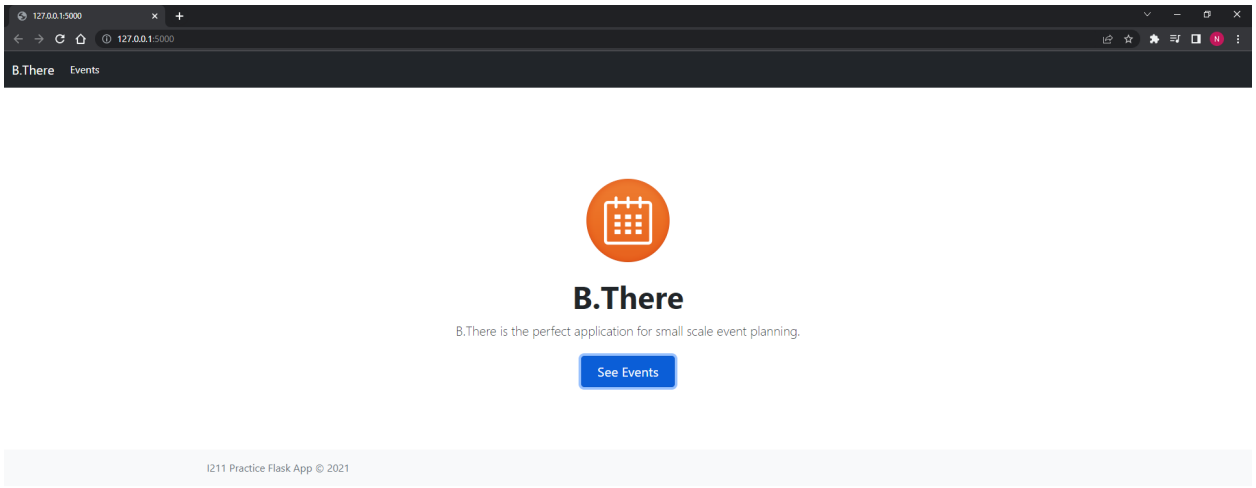
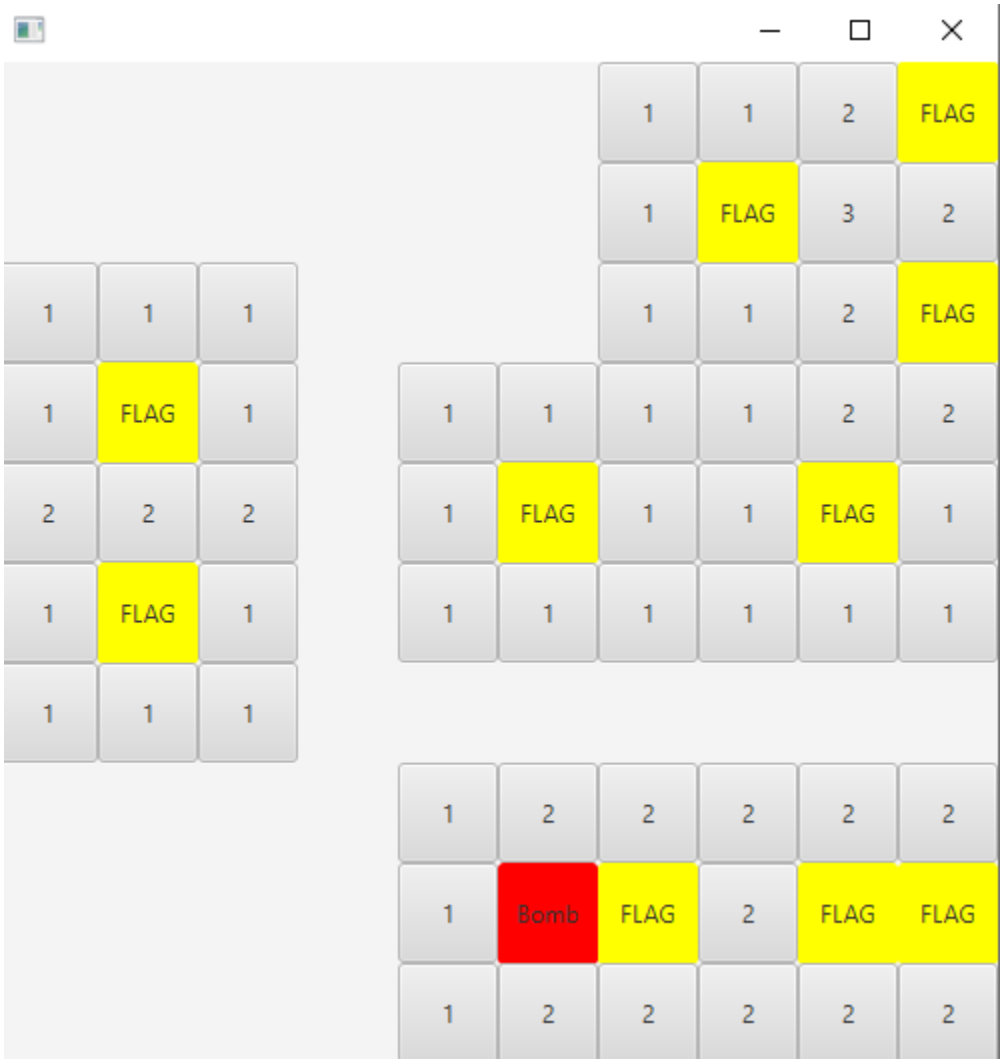
Anaconda Navigator / Spyder
Autograder

```
import pandas as pd

with open("grades18.csv") as g: #download grades file
    grades = pd.read_csv(g)
with open("zylabs19.csv") as z: #download whatever zylabs lesson
    zylabs = pd.read_csv(z)
lesson = input("Enter lesson: ") #enter given lesson

part = zylabs.columns[7] #figuring out point weighting
chal = zylabs.columns[8]
print(part)
print(chal)
lab = zylabs.columns[9]
print(lab)

try:
    partPoint = int(part[-3:-1])
except:
    partPoint = int(part[-2:-1])
try:
    chalPoint = int(chal[-3:-1])
except:
    chalPoint = int(chal[-2:-1])

totPoint = partPoint + chalPoint
print("Participation + Challenge =", totPoint)

gradeDict = {} #assigning practice and lab grades to email
for i, row in zylabs.iterrows():
    try:
        key = row["School email"].lower()
        practice = ((row[part] * partPoint) + (row[chal] * chalPoint)) / totPoint
        gradeDict[key] = [round(practice, 2), round(row[lab]/10, 2)]
    except:
        print("canvas grade export jank occured")


pracName = "" #figuring out exact names of column
labName = ""
for col in grades.columns:
    if "Lesson " + lesson + " Practice" in col:
        pracName = col
    if "Lesson " + lesson + " Lab" in col:
        labName = col


for i, row in grades.iterrows(): #updating grade csv
    try:
        studentName = row["SIS Login ID"].lower() + "@iu.edu" #matches email
        if studentName in gradeDict:
            row[pracName] = gradeDict[studentName][0]
            row[labName] = gradeDict[studentName][1]
        else:
            print(studentName, "failed to match") #if no match, they didn't do it, so 0
            #row[pracName] = 0
            row[labName] = 0
        grades.loc[i] = row #update row
```

Flask Web App Project Live web hosted content SQL and database merged application



I311 Java Project - Group Project Lead and Head Dev



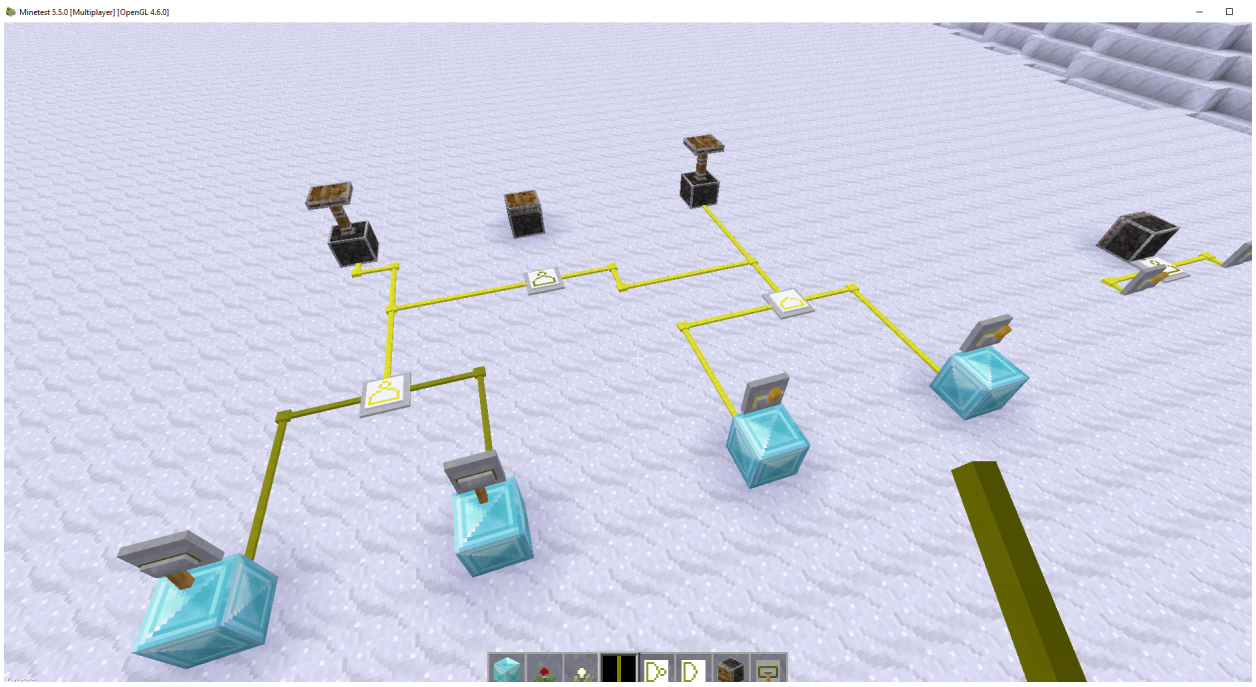
https://natcburk.pages.iu.edu/

Translation Application
https://docs.google.com/presentation/d/1EMMjus_WbuQww4zcUusmFs4DqkIJ_YPBj0trTfQ3vFI/edit#slide=id.p1

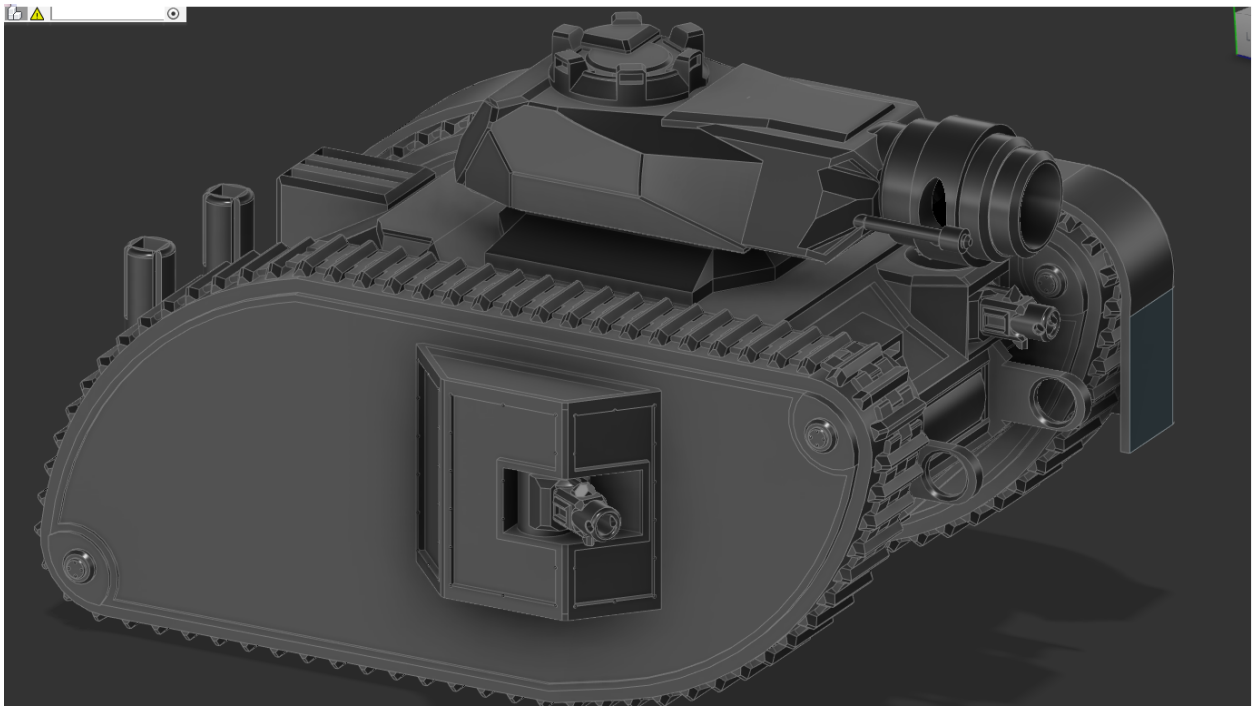
Track EZ Prototype and final portfolio
https://www.figma.com/file/pK1T487u4LtxBhqUBHXWJw/TrackEZ-High-Fidelity-Prototype

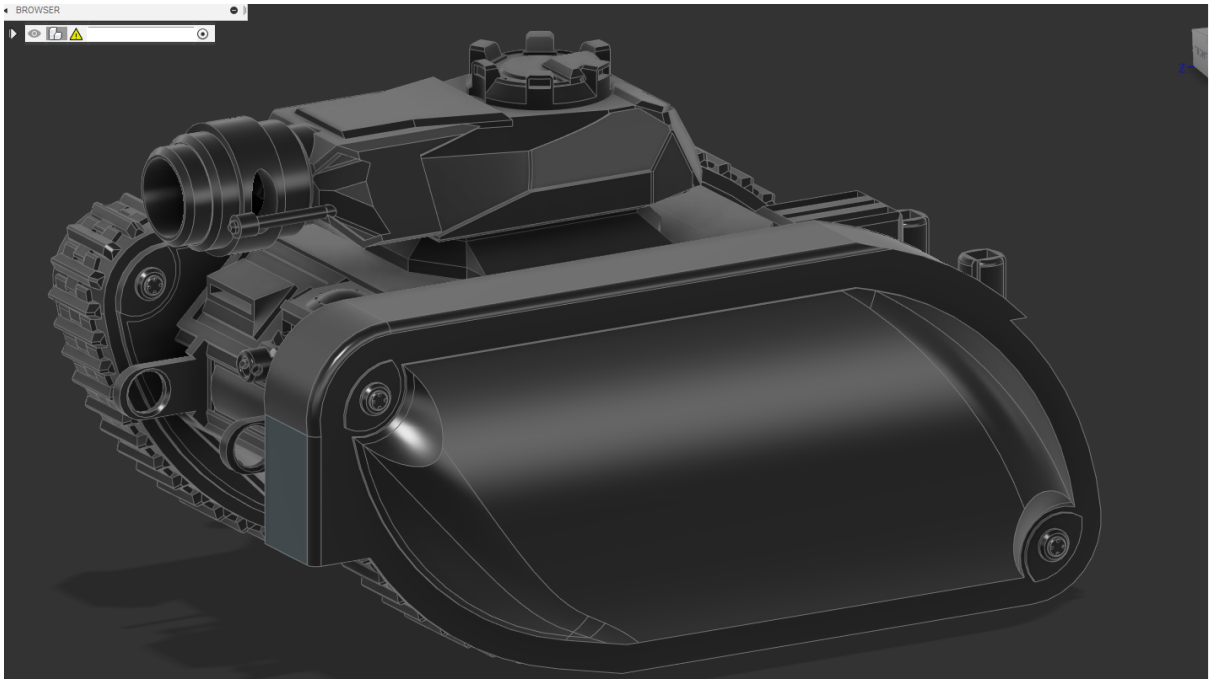
https://docs.google.com/presentation/d/1eezuXZjVvjfVCNVbzXsbl9dT3J6XJutzbP_znEILVEY/edit?usp=sharing
https://www.figma.com/file/GD1eClsgwuzany2EiSpozi/Untitled

Nand2Tetris Class Prototyping



3D Modeling, VR Asset creation





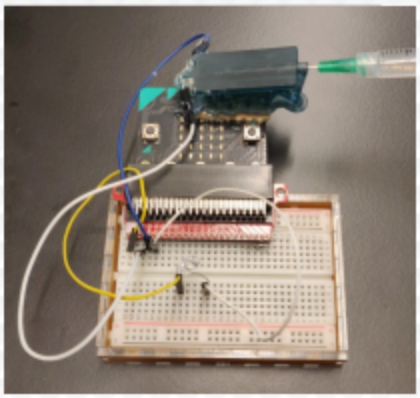
Microfluidic Cell Tester (In use in FAMES research laboratory)

Title: Microfluidic Cell Tester

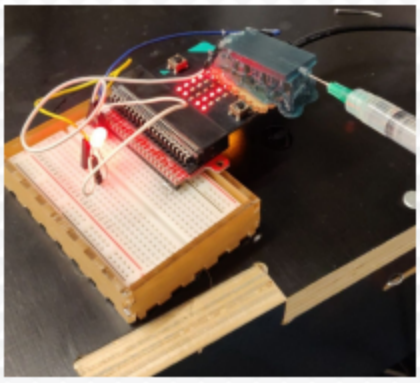
Project by Natalie Burke

Overview/Functionality

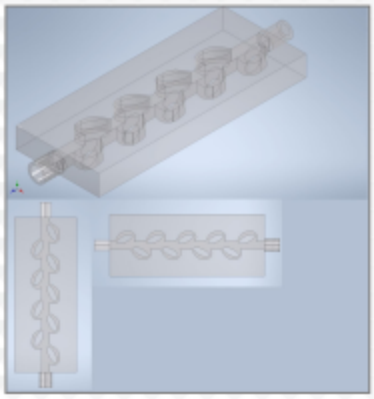
This Microfluidic Cell Tester allows the user to easily examine the physical properties of the internal structure of microfluidic cells. In this example, we have a microfluidic Tesla Valve - a valve that only allows fluid one-way travel due to unique geometry that forces the backpressure to stop flow. This part is designed for the FAMES lab, who are invested in this project. The Micro:bit is used to display the startup message, record the time undergone in the test of the microfluidic cell, and displays the time the trial lasted before a failure while in tandem triggering a visual alarm. Taking the volume of fluid used and length of test gives a direct answer to the pressures endured before a cell failure.



Objectives and Constraints



```
from microbit import *

startup = 0
counter = 0
while True:
    # Displays a start up screen, displaying that it is a fluid tester.
    if startup <= 0 :
        display.scroll('Fluid Tester')
        startup+=1
    # The pin is constantly reading if there is a voltage travelling through it.
    pin0.read_digital()
    counter +=1
    # Voltage is detected, indicating a failure
    if pin0.read_digital() > 0:
        pin1.write_digital(1)
        # Displays a failure message, in milliseconds
        display.scroll('FAIL millisec:')
        display.scroll(counter)
        # Displays the millisecond value a second time
        display.scroll("REP")
        display.scroll(counter)
        pin1.write_digital(0)
        # Resets start conditions and counter
        counter = 0
```



SCHOOL OF INFORMATICS, COMPUTING, AND ENGINEERING
INTELLIGENT SYSTEMS ENGINEERING