Step 5: Creating Fuzzy Membership Functions

In [2]:
```
pip install scikit-fuzzy
```

```
Defaulting to user installation because normal site-packages is not writeable
Collecting scikit-fuzzy
  Downloading scikit-fuzzy-0.4.2.tar.gz (993 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 994.0/994.0 kB 13.8 MB/s eta 0:00:0000:01
00:01
  Preparing metadata (setup.py) ... done
Requirement already satisfied: numpy>=1.6.0 in /home/student/s1292011/.local/lib/pyth
on3.10/site-packages (from scikit-fuzzy) (1.26.1)
Requirement already satisfied: scipy>=0.9.0 in /home/student/s1292011/.local/lib/pyth
on3.10/site-packages (from scikit-fuzzy) (1.11.3)
Collecting networkx>=1.9.0 (from scikit-fuzzy)
  Downloading networkx-3.2.1-py3-none-any.whl.metadata (5.2 kB)
Downloading networkx-3.2.1-py3-none-any.whl (1.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.6/1.6 MB 30.0 MB/s eta 0:00:00a 0:00:01
Building wheels for collected packages: scikit-fuzzy
  Building wheel for scikit-fuzzy (setup.py) ... done
  Created wheel for scikit-fuzzy: filename=scikit_fuzzy-0.4.2-py3-none-any.whl size=8
94079 sha256=4f8bc686e7f75f6ee32f4c8f9ffd3a28e65d25083f9851fdbb7f0c61fcf64257
  Stored in directory: /home/student/s1292011/.cache/pip/wheels/4f/86/1b/dfd97134a2c8
313e519bcebd95d3fedc7be7944db022094bc8
Successfully built scikit-fuzzy
Installing collected packages: networkx, scikit-fuzzy
Successfully installed networkx-3.2.1 scikit-fuzzy-0.4.2
Note: you may need to restart the kernel to use updated packages.
```

Step a: Insert the diagram containing the fuzzy membership functions

In [11]:
```
import numpy as np
import skfuzzy as fuzz
import matplotlib.pyplot as plt

so2_levels = np.arange(0, 0.25, 0.001)

# Adjust the "Good" membership function to start from 1 and decrease to 0
good = np.ones_like(so2_levels)
good[(so2_levels >= 0.020) & (so2_levels <= 0.040)] = 1 - np.linspace(0, 1, np.count

# Set the "Good" membership function to 0 for the rest of the range
good[so2_levels > 0.040] = 0

medium = fuzz.trimf(so2_levels, [0.021, 0.040, 0.100])
uh4sgs = fuzz.trimf(so2_levels, [0.041, 0.100, 0.120])
uh = fuzz.trimf(so2_levels, [0.101, 0.120, 0.150])
vuh = fuzz.trimf(so2_levels, [0.121, 0.150, 0.200])
h = fuzz.trapmf(so2_levels, [0.150, 0.200, 0.250, 0.250])

plt.figure(figsize=(10, 6))

# Plotting the adjusted "Good" membership function
plt.plot(so2_levels, good, label='Good')
plt.plot(so2_levels, medium, label='Medium')
plt.plot(so2_levels, uh4sgs, label='UH4SGs')
plt.plot(so2_levels, uh, label='UH')
plt.plot(so2_levels, vuh, label='VUH')
plt.plot(so2_levels, h, label='H')

plt.title('Fuzzy Membership Functions for SO2 Levels')
```

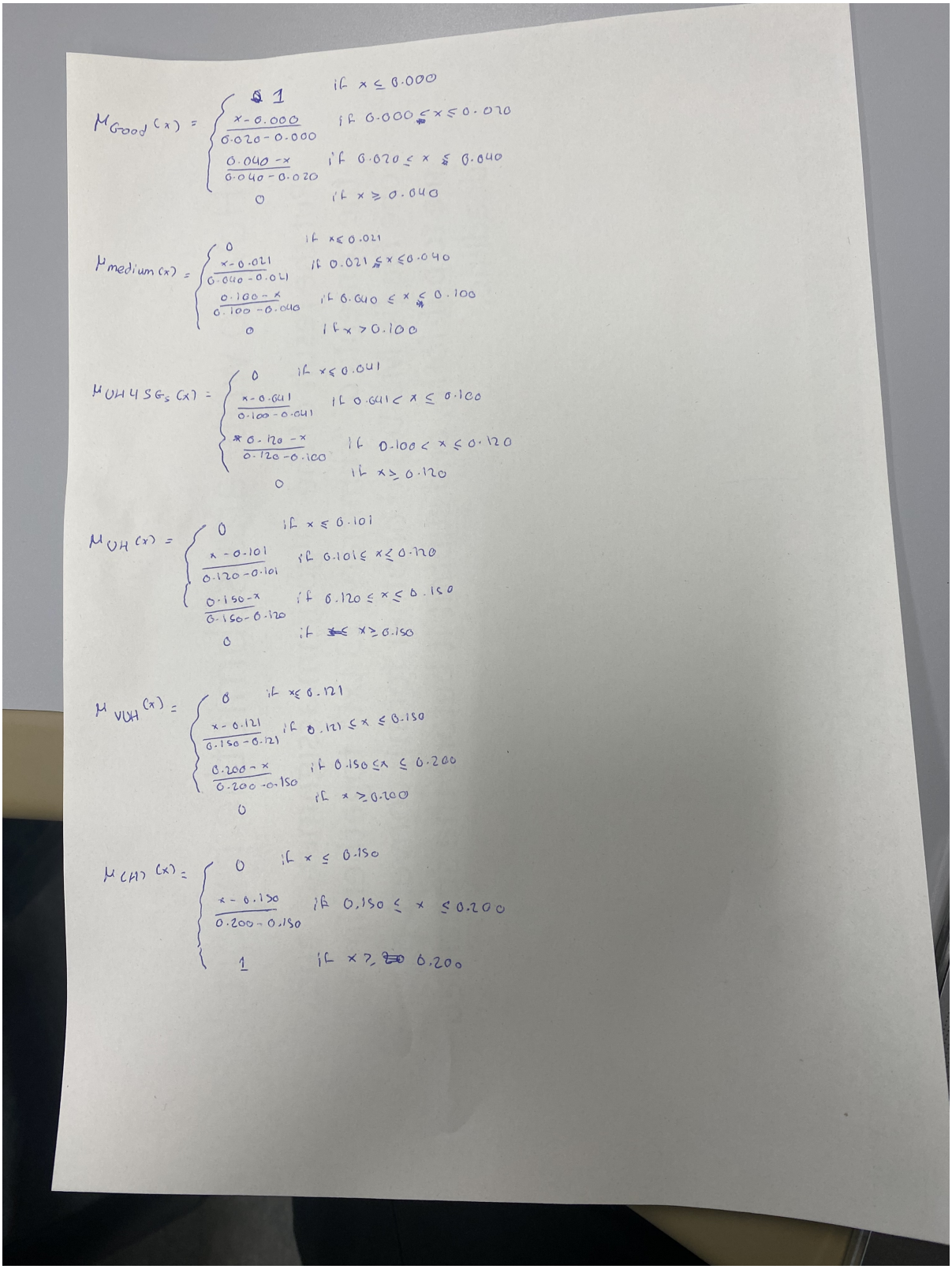
```
plt.xlabel('SO2 Levels (ppm)')
plt.ylabel('Membership Value')
plt.legend()
plt.show()
```

Step b: Insert the Picture containing the mathematical equations of the fuzzy membership functions

In [13]:
```
from IPython.display import Image, display

image_path = 'IMG_5495.jpg'

display(Image(filename=image_path))
```



Step 6: Mining Fuzzy Frequent Patterns

Step a: Read the file as a data frame

In [ ]:

Step b: Replace the cells containing the value 'None' to 'NaN'

In [ ]:

Step c: Do the preprocessing step by analyzing the data. In this step, replace the abnormal high and low SO2 values to zero.

In [ ]:

Step d: Convert the dataframe into a fuzzy transactional database.

In [ ]:

Step e: Extract patterns.

In [ ]:

In [ ]:
```
#Save the file as PDF and share it in the moodle.
```