COSC326 Etude 9 - Pulses Counting
Levi Schimanski
Student ID 6923634
Jada Mataroa
Student ID 9474013

For this etude we have a main method that receives a file from the arguments and reads the data into a list. This list is passed as a parameter into the pulses method which transforms this data into integer format and puts it into a list. It finds the mean of this data then plots it and calls the get_smooth method.

Our smoothing method removes data points that are lower than the mean as they may interfere with my averaging process. We do this by changing them to have the value of the mean and then averaging all of the data points with their 4 surrounding data points (2 on either side, the edge points are averaged using less points).

We find the mean of this smooth data and plot both the smooth data and the mean of it and then the program counts the peaks. To do this we find instances where 1 data point is above the smooth data's mean and then the next one is below it. Each of these instances is counted as a pulse and the number of pulses is incremented by one each time and then printed out at the end.

Our reasoning for smoothing was to merge pulses that were really close together so that they weren't counted as two pulses and to attempt to smooth out the random noise. Also we removed the low data points because they may have interfered with the high peaks we were interested in, because we didn't count the low troughs as pulses. We used the mean of the smooth peaks to judge whether a peak was a pulse because it is halfway up the data no matter the data and seemed like a good indicator. We used the two on either side of the data points when averaging because the pulses tended to occur once every 8 data points, meaning pulses that weren't meant to be merged weren't merged.