```
% Author: Spencer Goulette
% ECE 486 - HW #2 Problem 3
% February 5th, 2018

% Reads in audio at a samples per second of 44,100, Fs
% Samples are stored in x_long
% Samples from 55 seconds to 65 seconds are stored in x
[x_long,Fs] = audioread('guitar10min.ogg');
x = x_long(55*Fs:65*Fs,1);

% g1 relatively boosts the low frequency band below 441 Hz
% g2 relatively boosts the high frequency band above 441 Hz
% M is the number of samples to be averaged over
% D is the relative delay which is dependent on M
g1 = 42;
g2 = 1;
M = 151;
D = (M-1)/2;

% Total for summing in the running mean
total = 0;

% Temporarily assigning x to y1, y2, and y3
% Done to make sure y1, y2, and y3 are the same size as x
y1 = x;
y2 = x;
y3 = x;

% The running mean calculation runs through n times to get
% the appropriate number of elements for y1, y2, and y3,
% and to access every element in the x array. There then is another
% for loop that runs M times. If the indice n - k exists, then
% the element at that indice is added to the total. The total is
% then divided by M to get the average. This running average is
% y1. Y1 is then used to get y2. x is delayed for D samples and then
% subtracts y1 to create a highpass filter. Y1 is then multiplied by
% g1 and y2 is multiplied by g2. They are then added together to
 obtain
% y3, which is the result of the mixer. By changing g1, the low
% frequencies can be boosted. By changing g2, the high frequencies can
% be boosted.
for n = 1:length(x)
    for k = 1:M
        if (n - k) > 0
            total = total + x(n - k);
        end
    end
    y1(n) = total / M;
    total = 0;
    if (n - D) > 0
        y2(n) = x(n - D) - y1(n);
    else
```

```
        y2(n) = 0;
    end
    y3(n) = g1 .* y1(n) + g2 .* y2(n);
end

% (The player saturates for values of +/- 1)
y3 = y3/max(abs(y3));

% The new audio output is then played
% The original audio follows
obj1 = audioplayer(y3,Fs);
playblocking(obj1)
obj1 = audioplayer(x,Fs);
playblocking(obj1)
```

Published with MATLAB® R2017b