Project

- GitHub
  - ECE-251
    - .git
    - exam1
    - Exam2
      - .DS_Store
      - Dictionary1.txt
      - e1
      - e1.c
      - e1.pdf
      - e2
      - e2.c
      - e3
      - e3.c
      - exam 2.3 file and output.png
      - Screen Shot 2018-11-04 at 10.16.5
      - test.c
    - project1
    - project2
    - project3
    - project4
    - project5
    - project6
    - project7
    - project8
    - project9
    - project10
    - project11
    - project12
    - project13
    - project14
    - project15
    - project16
    - project17
    - project18
    - .DS_Store
    - README.md

e2.c | e3.c | e1.c

```c
#include <stdio.h>
#include <math.h>
#define NUMLEVELS 40
#define NUMVALS 50

int main(){
  float amplitude;
  float stepSize;
  float lowerLimit, upperLimit;
  float a;
  float b;
  float c;
  float v;
  float t;
  float x;
  float timeRange;
  float levelNum;
  a = .25; //coefficent for x^2
  b = -40; //coefficent for x
  c = 4; //constant
  amplitude = 2000; //amplitude
  timeRange = 200; //timeRange
  float xStep;
  xStep = timeRange/NUMVALS;


  stepSize = amplitude/NUMLEVELS*2;

  for(levelNum=0; levelNum<=NUMLEVELS; levelNum++){ //main loop
    //printf("%.1f\n", stepSize*levelNum-amplitude);
    upperLimit = amplitude-levelNum*stepSize; //calculates upperLimit
    printf("%.1f|\n",upperLimit);
    lowerLimit = upperLimit-stepSize; //calculates lowerLimit
    //printf("lowerLimit: %.1f\n", lowerLimit);
    for(t =0; t<NUMVALS;t++){ //loop through x
      x = xStep*t; //calculate x value
      v = a*pow(x,2)+b*x+c; //calculate voltage
      //printf("v: %.1f \n", v);
      if((v>=lowerLimit) && (v<=upperLimit)){ //in between
        printf("*"); //print *
      }
      else if((upperLimit>0) && (lowerLimit<0)){ //zero
        printf("-"); //print -
      }
      else{
        printf(" "); //otherwise print nothing
      }
    }
    printf("\n"); //print a new line

  }
  return 0;
}

```