Name: Hadi Bagdadi
Partner: Jamal Nawabi

This week we completed our project which was to construct a working calculator using Arduino Mega. We constructed a calculator by using a 16x2 LCD and a 4x4 keypad with Arduino. We constructed a program that calculates the user inputs, then outputs the user input and result onto the LCD. One of the concepts learned from class that we applied in our project was a truth table that we constructed for all possible keypad inputs as seen below. I personally figured out how to backlight the display so that the display is more readable. I am currently working on debugging the code, starting up the calculator with a buggy display that corrects itself after a calculation.

Materials: Arduino Mega; 16x2 LCD; 4x4 Keypad; 9V Battery; Breadboard; Jumper cables

Variables:

```
const byte ROWS = 4; // Four rows

const byte COLS = 4; // Three columns

const char ADD = 'A';

const char SUB = 'B';

const char MUL = 'C';

const char DIV = 'D';

const char CLEAR = '*';

const char EQUAL = '#';

byte rowPins[ROWS] = { 0, 1, 2, 3 };// Connect keypad ROW0, ROW1, ROW2 and ROW3 to these Arduino pins.

byte colPins[COLS] = { 4, 5, 6, 7 }; // Connect keypad COL0, COL1 and COL2 to these Arduino pins.

Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS ); // Create the Keypad

const int rs = 8, en = 9, d4 = 10, d5 = 11, d6 = 12, d7 = 13; //Pins to which LCD is connected

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

long Num1,Num2,Number;

char key,action;

boolean result = false;
```

Name: Hadi Bagdadi
Partner: Jamal Nawabi

2D array:

```
// Define the Keymap

char keys[ROWS][COLS] = {

{'1','2','3', ADD},

{'4','5','6', SUB},

{'7','8','9', MUL},

{CLEAR,'0',EQUAL, DIV}

};
```

Functions:

```
void setup()

void loop()

void DetectButtons()

void CalculateResult()

void DisplayResult()
```

Name: Hadi Bagdadi
Partner: Jamal Nawabi

Truth Table:

a0	a1	a2	a3	a4	a5	a6	a7	O	Symbol
1	2	3	4	5	6	7	8		
1	0	0	0	1	0	0	0	1	1
1	0	0	0	0	1	0	0	2	2
1	0	0	0	0	0	1	0	3	3
1	0	0	0	0	0	0	1	A	ADD +
0	1	0	0	1	0	0	0	4	4
0	1	0	0	0	1	0	0	5	5
0	1	0	0	0	0	1	0	6	6
0	1	0	0	0	0	0	1	B	SUB -
0	0	1	0	1	0	0	0	7	7
0	0	1	0	0	1	0	0	8	8
0	0	1	0	0	0	1	0	9	9
0	0	1	0	0	0	0	1	C	MUL *
0	0	0	1	1	0	0	0	*	CLEAR
0	0	0	1	0	1	0	0	0	0
0	0	0	1	0	0	1	0	#	EQUAL =
0	0	0	1	0	0	0	1	D	DIV /