**Project Option 1: Patterns in Expanding Letters:**

Choose three letters of the alphabet (maybe your initials?). For each one, design a way to make the letter out of squares. Make a sequence of enlarged versions of that letter (make them look pretty-ish), and then find a formula for the number of squares that make up the *n*th term in the sequence.

For example, here’s my first initial ‘A’ as enlarging letters made of squares.

**Project Option 2: Staircase Patterns**

Consider the following sutation: a staricase is made from Legos bricks. Each time a new brick is added, it is offset by one row of studs.

1. Pick any lego size brick (other than a cube). Record it’s volume and surface area.
2. Build staircases of different sizes with that size Lego brick. Each time you build a staircase, record the number of bricks used, the volume of the staircase, and the surface area of the staircase.
3. What patterns do you find in your data? What do you predict will be the volume and surface area of the staircase that has 10 bricks? 25 bricks? *n* bricks?
4. How do the patterns change if you build a staircase from bricks of a different length?

**Project Option 3: BASIC Solution of Inequalities**

Write a BASIC program on your graphing calculator that will solve inequalities of the form:
*ax + b > c*. Your program should ask the user for the values of *a*, *b*, and *c*, and then print an appropriate solution. You will probably need to use an IF-THEN statement in your program. This statement allows the computer to make a choice, depending on what values of *a*, *b*, and *c* are given. You may also need to determine how to use the INPUT statement and the PRINT statement on your computer. Test your program to make sure it works for all values of *a*, *b*, and *c*.