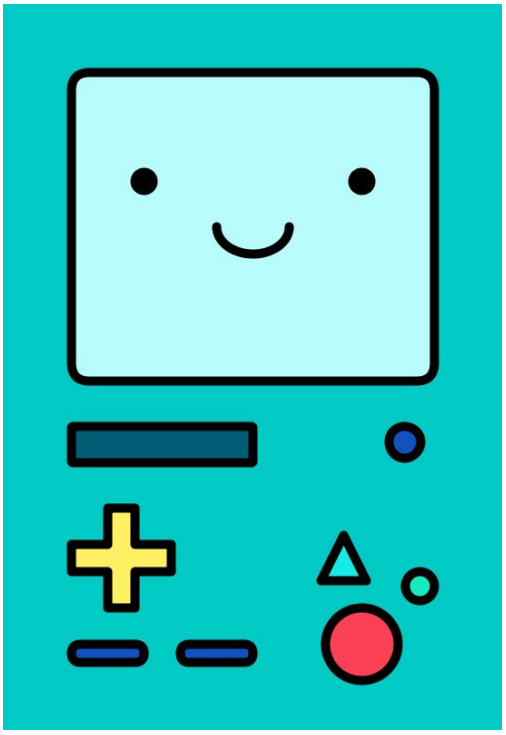
Bonus points for all of the below if you embed them as a functioning sketch within your blog!

1. **Drawing**
   Using primitive shapes (rect, line, ellipse, etc.), recreate the image below. Note that you'll need beginShape() and endShape() for the D-pad:



2. **Interaction**
   Make 3 different drawing programs
   a. Create a black circular "brush" that follows your mouse
   b. Create a line from where your mouse was in the last frame to where your mouse is in the current one and vary the thickness of that line according to how fast you moved the mouse.
   c. Make a black square at the location of your mouse every time you click, click and drag to draw a line with a square brush, and erase the canvas when you press a key (set rectMode() to center).

a. b. c.

3. **Variables**
    a. Make 3 bubbles of random diameter (between 3 and 30 pixels) appear at random locations on the x axis at the bottom of the screen and then float up and off the screen while wiggling to the left and right a bit.

4. **Conditionals & logical operators**
    a. Set the screen to white and divide it into 4 quadrants with black lines. Turn the quadrant that the mouse is in red in a rollover fashion
    b. Make a ball bounce around the screen, changing direction when it hits the edges. (Try moving background() from setup to draw and vice versa and see what happens).
    c. Make a rectangle in the middle of the screen. Click on the rectangle to change the colour of the rectangle from whte to black.
    d. Make a small square appear on the screen and drop towards the bottom. When it hits the bottom, have it bounce, but with gravity affecting it, such that each time it bounces, it bounces less high.
    e. Make a circle that shudders with terror every time you point the mouse at it. Use dist().
    f. Make a circle that flees to a random location every time you point the mouse at it. Use dist().

5. **Loops**
    a. While loop. Draw short vertical lines spaced at 20 pixels apart as far as 300 pixels across the screen.
    b. Alter the code from a. to change the spacing from 20 pixels to a random number between 15 and 25 every frame.
    c. Alter the code from b. and use a for loop instead of a while loop.
    d. Using nested for loops, create a grid of squares. Create an effect where the closer each square is to the mouse, the lighter its shade. Use dist().

6. **Functions & objects**
    a. Draw a smiley face within a function, and use the function to display multiple smiley faces at different locations on the screen simultaneously.
    b. Alter the code you wrote in 3.a. Write functions for displayBubble() and moveBubble() and wiggleBubble. Just code for 1 bubble rather than 3.
    c. Create class for Bubble containing the functions you wrote in 6.b. Create 3 Bubble objects in your main code with random starting points on the x axis and random widths for the bubbles.

7. **Arrays**
    a. Using the class Bubble you made in 6.c. use an array to make 50 bubbles appear on screen.
    b. Using and ArrayList make a bubble appear on screen every time you click the mouse and make one disappear every time you press a key.