Hi there:)

I’ve completed the coding task but wanted to mention a couple of things. It was a fun test by the way, especially when looking for edge cases:)

**Parser code that should really be in a method**

As the code reads from a command file I though I’d create a block of code to act as a parser for it. I wrapped this logic in a method (which would be both the tidiest and most readable option, but unfortunately couldn’t get it to work as I had to call the method within another method (def run) but couldn’t get it to run with the ‘bin/bitmap\_editor …’ command line, I’m guessing that’s because it doesn’t create a bitmap\_editor instance, so cannot call the ‘def parser’ method I created. So I’ve had to leave the code in there as the first switch statement. If you scoot past this the functional code is a bit lower down.

**Test code**

I wanted to create a test block to pull the test cases into (the test cases are in the file examples/error\_test.txt), but had trouble with handling input, as the code is set up for a file to be read.

The parser is written to capture all the error conditions I could envisage, so hopefully I’ve covered them all (I’d be really interested to know if I missed any!):

I # Too few args

I 4 # Too few args

I 4 5 6 # Too many args

I 4 A # Incorrect arg value – non-numeric

I B 3 # Incorrect arg value – non-numeric

I 0 4 # Incorrect arg value – zero

I 3 0 # Incorrect arg value – zero

I 5 5 #! *Correct input – defines grid for tests below*

C 1 # Too many args

L 1 # Too few args

L 1 2 # Too few args

L 3 4 5 6 # Too many args

L A 2 B # Incorrect arg value – non-numeric

L 1 A B # Incorrect arg value – non-numeric

L 0 1 B # Incorrect arg value – zero

L 1 0 B # Incorrect arg value – zero

L 5 6 B # Out of range arg (Y axis)

L 6 5 D # Out of range arg (X axis)

V 1 # Too few args

V 1 2 3 A B # Too many args

V 1 6 5 A # Y axis start arg out of range

V 1 3 6 A # Y axis end arg out of range

V A 2 3 A # Incorrect arg value – non-numeric

V 1 A 3 A # Incorrect arg value – non-numeric

V 1 2 A A # Incorrect arg value – non-numeric

V 0 2 3 A # Incorrect arg value – zero

V 1 0 3 A # Incorrect arg value – zero

V 1 2 0 A # Incorrect arg value – zero

V 6 2 4 A # Out of range arg (X axis)

V 2 5 4 A # Y axis start larger the Y axis finish point

V 2 5 6 A # Y axis end point out of range

V 2 6 5 A # Y axis start point out of range

V 2 5 5 A # Start and end values for Y axis line the same (creates a point, not a line)

H 1 # Too few args

H 1 2 3 A B # Too many args

H A 2 3 A # Incorrect arg value – non-numeric

H 1 A 3 A # Incorrect arg value – non-numeric

H 1 2 A A # Incorrect arg value – non-numeric

H 0 2 3 A # Incorrect arg value – zero

H 1 0 3 A # Incorrect arg value – zero

H 1 2 0 A # Incorrect arg value – zero

H 1 5 3 A # X axis start point larger that end point

H 1 5 5 A # Start and end values for X axis line the same (creates a point, not a line)

H 2 4 6 A # X axis end point out of range

H 2 6 4 A # X axis start point out of range

H 6 4 5 A # Y axis out of range

S 1 # Too many args

X # Unknown command