```
1   //
2   // Digital Logic Display & Number Lab
3   // Part 1
4   // Neil Nie
5   // (c) Oct 28th, 2018.
6   //
7
8   module seven_segment_display (
9
10     in,          // accepts a four bit input
11     out          // returns a six bit output
12  );
13
14  input [3:0] in;   // A B C D
15  output [6:0] out; // D C B A G
16
17  // waiting to be implemented
18
19  assign out[6] = (~in[3] & ~in[2] & ~in[1])    + (~in[3] & in[2] & in[1] & in[0]);
20  assign out[5] = (in[3] & in[2] & ~in[1])      + (~in[3] & in[1] & in[0]) + (~in[3] & ~in[2] &
     in[0]) + (~in[3] & ~in[2] & in[1]);
21  assign out[4] = (~in[3] & in[0])              + (~in[3] & in[2] & ~in[1]) + (~in[2] & ~in[1]
    & in[0]);
22  assign out[3] = (in[2] & in[1] & in[0])       + (~in[3] & in[2] & ~in[1] & ~in[0]) + (in[3] &
     ~in[2] & in[1] & ~in[0]) + (~in[2] & ~in[1] & in[0]);
23  assign out[2] = (in[3] & in[2] & ~in[0])      + (~in[3] & ~in[2] & in[1] & ~in[0]) + (in[3] &
     in[2] & in[1]);
24  assign out[1] = (in[2] & in[1] & ~in[0])      + (in[3] & in[1] & in[0]) + (~in[3] & in[2] & ~
    in[1] & in[0]) + (in[3] & in[2] & ~in[0]);
25  assign out[0] = (in[2] & ~in[1] & ~in[0])     + (in[3] & in[2] & ~in[1]) + (~in[3] & ~in[2] &
     ~in[1] & in[0]) + (in[3] & ~in[2] & in[1] & in[0]);
26
27  endmodule
28
29  module Part1(
30
31     SW,
32     HEX0,
33  );
34
35  input [3:0] SW;
36  output [6:0] HEX0;
37
38  wire [6:0] out;
39
40  seven_segment_display (SW, out);
41
42  assign HEX0 = out;
43
44  endmodule
45
```