adder.sv handout

```
module adder_4_bit(output c_out, output [3:0] sum,
     input c_in, input [3:0] a, b);

  // uses 4 instances of the full_adder module
  wire [3:1] carry;
  full_adder fa0(.c_out(      ), .sum(      ),
     .c_in(      ), .a(      ), .b(      );
  full_adder fa1(.c_out(      ), .sum(      ),
     .c_in(      ), .a(      ), .b(      );
  full_adder fa2(.c_out(      ), .sum(      ),
     .c_in(      ), .a(      ), .b(      );
  full_adder fa3(.c_out(      ), .sum(      ),
     .c_in(      ), .a(      ), .b(      );
endmodule


module adder_4_bit_gen(output c_out, output [3:0] sum,
     input c_in, input [3:0] a, b);

  // using generate to avoid repetition
  wire [4:0] carry;
  generate
    genvar i;
    for(i=0; i<4; i=i+1) begin: adder
      full_adder fa(.c_out(     ), .sum(     ),
           .c_in(      ), .a(      ), .b(      );
     end
  endgenerate

  assign carry[0] =       ;
  assign c_out =       ;

endmodule
```