```
1  --------------------------------------------------------------------------------
2  -- Name: Capt Jeff Falkinburg
3  -- Date: Spring 2016
4  -- Course: ECE 281
5  -- File: ce2_ha_structural.vhd
6  -- HW:   Lecture 13
7  -- Purp: Half Adder - Structural Implementation.
8  --
9  -- Doc:  None
10 -- Academic Integrity Statement: I certify that, while others may have
11 -- assisted me in brain storming, debugging and validating this program,
12 -- the program itself is my own work. I understand that submitting code
13 -- which is the work of other individuals is a violation of the honor
14 -- code.  I also understand that if I knowingly give my original work to
15 -- another individual is also a violation of the honor code.
16 --------------------------------------------------------------------------------
17 library IEEE;                  -- These lines are similar to a #include in C
18 use IEEE.std_logic_1164.all;
19 library unisim;                -- Use these libraries if you are using primitive
   components
20 use unisim.vcomponents.all;
21
22 entity ce2_ha_structural is
23     Port ( A : in  STD_LOGIC;
24            B : in  STD_LOGIC;
25            S : out  STD_LOGIC;
26            Cout : out  STD_LOGIC);
27 end ce2_ha_structural;
28
29 architecture Structural of ce2_ha_structural is
30    component AND2
31       port ( i0, i1  : in std_logic;
32              o       : out std_logic);
33    end component;
34    component XOR2
35       port ( i0, i1  : in std_logic;
36              o       : out std_logic);
37    end component;
38    signal   s1, s2 : std_logic;  -- wires which begin and end in the component
39 begin
40    unit1:   AND2
41    port map (  -- s1 <= A AND B; (i.e. Cout)
42       i0 => a,
43       i1 => b,
44       o  => s1);
45    unit2:   XOR2
46    port map (  -- s2 <= A XOR B; (i.e. Sum)
47       i0 => a,
48       i1 => b,
49       o  => s2);
50
51    Cout <= s1;
52    S <= s2;
53
54 end Structural;
```