```
25  library ieee;
26  use ieee.std_logic_1164.all;
27  use ieee.numeric_std.all;
28
29  package insbuffer_type is
30  type insBuffer is array(63 downto 0) of std_logic_vector (24 downto 0);
31  end package insbuffer type;
32  use work.insbuffer type.all;
33
34  library ieee;
35  use ieee.std logic 1164.all;
36  use ieee.numeric std.all;
37  use IEEE.STD_LOGIC_UNSIGNED.ALL;
38
39  entity ins_buffer is
40      port(
41      instructions : in insBuffer;
42      clk : in std_logic;
43      ins out : out std_logic_vector(24 downto 0)
44      );
45  end ins buffer;
46
47  --}} End of automatically maintained section
48
49  architecture behavioral of ins_buffer is
50  signal PC : std_logic_vector(5 downto 0) := "000000";
51  begin
52      process(clk, instructions)
53      begin
54          if rising_edge(clk) then
55              ins out <= instructions(to integer(unsigned(PC)));
56              PC <= PC + std_logic_vector(to unsigned(1, PC'length));
57          end if;
58      end process;
59
60       -- enter your statements here --
61
62  end behavioral;
```