```
1   // Eugene Ngo
2   // 5/3/23
3   // EE 469
4   // Lab 3
5
6   /* imem is the read only, 64 word x 32 bit per word instruction memory for our processor.
7   ** Its module is written in RTL, and it strongly resembles a ROM (read only memory) or LUT
8   ** (look up table). This memory has no clock, and cannot be written to, but rather it
9   ** asynchronously reads out the word stored in its memory as soon as an address is given.
10  ** The address and memory are byte aligned, meaning that the bottom two bits are discarded
11  ** when looking for the word. One important line to note is the
12  **      Initial $readmemb("memfile.dat", memory);
13  ** which determines the contents of the memory when the system is initialized. You will alte
14  ** this line to use programs given to you as a part of this lab.
15  */
16
17  // addr - 32 bit address to determine the instruction to return. Note not all 32 bits are
    used since this
18  //        memory only has 64 words
19  // instr - 32 bit instruction to be sent to the processor
20  module imem(
21      input  logic [31:0] addr,
22      output logic [31:0] instr
23  );
24      logic [31:0] memory [63:0];
25
26      // modify the name and potentially directory prefix of the file within to load the
    correct program and preprocessing
27      initial $readmemb("C:\\Users\\egeen\\Desktop\\School\\EE 469\\Lab\\Lab 3\\memfile.dat",
    memory);
28
29
30      assign instr = memory[addr[31:2]]; // word aligned, drops bottom 2 bits
31
32  endmodule
```