1. (8 points) For each of the described program behaviors below, what is the Average Memory Access Time, Average Stalls Per Instruction, and Average CPI. In all cases, assume the following penalties:
   Branch Mispredict - 1 cycle; L1 Hit - 1 cycle; L2 Hit - 20 cycles; L2 Miss - 400 cycles
   a. 10% branches, 20% mispredicted. 30% Accesses, 3% miss L1 (hit L2), 2% miss L2
   b. 5% branches, 50% mispredicted. 50% Accesses, 5% miss L1 (hit L2), 1% miss L2
   c. 30% branches, 10% mispredicted. 50% Accesses, 5% miss L1 (hit L2), 4% miss L2

2. Consider changing a cache from direct-mapped to 2-way set associative without otherwise altering it.
   a. (2 points) Can this change increase the AMAT? Why or why not?
   b. (2 points) If not, is the AMAT guaranteed to decrease? Why or why not?

3. For the following question, assume the cache line is at least 16 bytes. Also assume that you’re using a compiler that doesn’t make any optimizations of this type.

   You’re working on a program which randomly accesses elements in a large array of 12-byte structures. Every field in a structure is accessed before the program moves to another one.

   Another programmer mentions hearing something about structures being powers of two, and suggests you add padding to the end of the structure to increase it to 16 bytes.

   a. (4 points) Could this increase performance? Why?
   b. (4 points) Could this decrease performance? Why?
   c. (5 points towards quiz 5) If instead the structure were 20 bytes, would padding it up to 32 be likely to affect performance in the same way as it did for the 12 to 16 change?