(If you need to any assumptions to solve the following problems, please write them clearly before writing the answers.)

1. Write ARM assembly code to implement the following conditional: (10 pts)

        if (x-y <3) {a= b-c; x=0;}
        else {a=c-b; y=0;}

2. Write ARM assembly for the following C code. (20 pts)

    int foo (int x1, int x2) { return x1 + x2; }

    void scum (int r) {
      for (i=0; i=2 ;i++)
        foo (r +i, 5);
    }
    main () {
      scum (3);
    }

3. The following code is executed by an ARM processore with each instruction executed exactly once:

```
                MOV r0, #0      ;
                LDR r1, #10     ;
                MOV r2, #0      ;
                ADR r3, c       ;
                ADR r5, x       ;
        loop    CMP r0, r1      ;
                BGE loopend     ;
                LDR r4, [r3, r0];
                LDR r6, [r5, r0];
                MUL r4, r4, r6  ;
                ADD r2, r2, r4  ;
                ADD r0, r0, #1  ;
                B    loop
```

Show the contents of the instruction cache for these configurations, assuming each line hold one ARM instruction: (30 pts)

(a) direct mapped cache with four lines

(b) direct mapped cache with eight lines

(c) two-way set-associative cache with four line per set

4. Explain in detail all possible cases that can occur during an instruction fetch and draw its algorithmic flow chart including page fault, cache miss, and TLB miss. (15 pts)

5. Explain each of the following terminologies. (15 pts)
   (a) compulsory miss
   (b) capacity miss
   (c) conflict miss

6. Build a FSM for each of the following. (20 pts)
   (a) an AMBA master interface
   (b) an AMBA slave interface