IA-64 Homework

Problem 1
Write the IA-64 code using predication instead of branches to execute the mathematical expression X = |A-B|. Pseudocode for this expression is given below:

if (A >= B)
    X = A - B
else
    X = B - A

Problem 2
Assume that a load takes 5 cycles to complete and all other instructions can be completed in 1 cycle. Schedule the following code to minimize the latency and maximize the parallelism. Be sure to use speculation and/or advanced loads when necessary. Indicate which instructions can be executed in parallel.

```plaintext
add         r0 = r0,10
Label 1    ld8    r1 = [r15],8
add         r2 = r1, r2
sub         r0 = r0,1
cmp.eq      p1 = r0,0
(p1)br.cond Label 1
cmp.neq     p2 = r2,r12
(p2)br.cond Label 2
ld8         r3 = [r7]
Label 2    add         r3 = r2,r4
             st8          [r7] = r3
```

Problem 3
Software pipeline the loop below for the IA-64. To do this you will need to
1. initialize LC, EC, etc,
2. assign registers (both general and predicate),
3. convert from ‘C6000 instructions to IA-64 instructions (see class example).

Assume the loop is executed 6 times.

```plaintext
loop:
    ldh      *a_m++, a
    ldh      *b_m++, b
    add      a, b, c
    mpy      c, prod, prod
    [i]      sub      i,1,i
    [i]      b      loop
```

Work this problem two ways. The first way assumes the results of each instruction appears the next cycle. The second way assumes that the load have the same delay that the ‘C6000 has (i.e. 4 nops after a load).