Purpose:  Use Threading techniques to solve a problem  
  
This lab models a Robot.  The Robot executes the commands between the <Command>byte1byte2</Command> tag.  
  
Byte1  
0 1 = left  
1 2 = right  
2 4 = up  
3 8 = down  
4 16 = clockwise  
5 32 = counterclockwise  
6 64 = fast  
7 128 = slow  
Byte2  
0 - 255 Degrees of rotation  
  
Data File Excerpt:  
<Motion>  
<Arm4>  
<Command>H'<\Command> // 01001000 00101110 = Fast Up 39 degrees  
<\Arm4>  
<\Motion>  
<Motion>  
<Arm2>  
<Command>B€<\Command>  
<\Arm2>  
<\Motion>  
  
This Robot has 4 robotic arms.  Model these arms with a Threaded QUEUE:  
Arm1 = Q[0] (Thread0)  
Arm2 = Q[1] (Thread1)  
Arm3 = Q[2] (Thread2)  
Arm3 = Q[3] (Thread3)  
  
The Robot can only execute one command at a time.  Each arm is available to execute any <ArmX> command in any order:  
\* a command is placed in a Thread  
\* place the Thread in the Thread Queue  
\* when the Robot is available, the Thread Queue passes it a command to execute  
\* if the Thread Queue is full (all 4 threads have waiting commands), the Thread queue waits until a queue position opens.  
  
Process the XML data file (Robot.xml), each line represents 1 command to the Robot.  To simulate Robot action, the executes the command by:  
\* Determining the speed (fast or slow)  
\* Determining the degrees (1 - 255)  
\* Put the current Thread to sleep for #Degrees \* 1 (fast) or 2 (slow) milliseconds  
\* Output the robot command as part of the simulation (i.e. Fast Up 39)