1. Intro
   1. Goal
      1. Develop a sheet music recognition system for scanned/electronic music scores as good as possible within the time limit
   2. Motivation
      1. Learning tool
      2. Composition – to have a baseline and rearrange for other instruments
      3. Enjoyment – what an arrangement sounds like
      4. Personal connection – manually inputting scores
   3. Users
      1. Instrument learners
      2. Composers
   4. Software
      1. Java program
      2. User interface – progress displayed
   5. Limitations
      1. No proper binarization for scanned scores
      2. Assuming 300 dpi
      3. No hand-written scores/old-fashioned notations
      4. Only some music notation
2. Whole project
   1. Electronic music (Entertainer)
      1. Stave removal
         1. Histogram, threshold
         2. Don’t remove if something above/below
      2. Vertical line removal
         1. Find black pixel
         2. Find width and height
         3. Bar line recognition
         4. Remove line
            1. Don’t remove if something to the left or right
      3. Connected component analysis
         1. Scan image with a mask
         2. Assign labels
         3. Record label equivalence
         4. Replace labels with their representative label
         5. Draw bounding boxes
      4. Symbol recognition
         1. Decision tree
         2. Initial recognition (beams, dots, tails, ties, slurs)
         3. Further recognition (notes, accidentals, rests)
            1. Loop through each system
            2. Loop through each measure
            3. Loop through each stave
         4. For notes and rests, look for modifying symbols (dot, beam, tail, tie, slur)
         5. Record a note
            1. Determine pitch based on vertical position
            2. Put into a chord if multiple notes with the same x position
         6. Convert everything into MusicXML
   2. Scanned music (Auld Lang Syne)
      1. Deskewing
         1. SGD – rotate by a small angle until error increasing
         2. Histogram
         3. Function – threshold, add values
      2. Binarization
      3. Line removal
         1. Remove extra pixels to handle distortion
      4. Patching
         1. Because of broken symbols, especially minims and semibreves
3. Project management
   1. Logbook of ideas (including rejected ones)
   2. Literature review before every stage
   3. Git repository – single features after testing
   4. Systematic work every week
   5. Skip difficult stages to develop a complete system
   6. If more time, would do:
      1. More complex notation – chords, dynamics, glissandos
      2. Binarization, better rotation (shearing) and line removal (Hough transform)
      3. No vertical line removal
   7. If again, would’ve focussed on developing the first prototype more