**Sequencing Rationale**

Dawn Bryant

There are a few different sequencing methods I would use to implement this phonics-based reading curriculum. Some skills can be taught concurrently, as they do not necessarily build on each other, but work together to make connections in the brain. Of course, the letter-sound identification must be learned before students can look at a word and decode it. However, while students are learning the letters and sounds, they can also be learning phoneme segmentation skills and rhyming skills. The students can be learning these skills at the same time and making connections and relations in their brains. One day’s reading lesson could include about fifteen minutes of letter-sound relations instruction, about ten minutes of phoneme segmentation instruction, and about ten minutes of rhyming instruction.

There are some skills that are needed as a prerequisite for more complicated skills in the phonics-based program. A student must have knowledge of letter-sound relations before he or she can blend sounds together or decode words. Therefore, I would use the logical prerequisite sequence when planning instruction. Letter-sound identification and phoneme segmentation are logical prerequisites for blending sounds and decoding text. Blending sounds is a logical prerequisite for decoding text.

Therefore, my curriculum would include a sequence of letter-sound identification, phoneme segmentation, and rhyming skills to be taught together. Once a student has mastered letter-sound identification, he or she can begin instruction in blending sounds. Mastery of phoneme segmentation is not necessary prior to beginning instruction in blending sounds. The skill of blending sounds must be acquired before a student can begin instruction in decodable text. Phoneme segmentation and rhyming instruction can continue throughout the higher level lessons until mastery occurs.

Letter-Sound Identification Phoneme Segmentation Rhyming

Blending Sounds *continue until mastery continue until mastery*

Decodable Text