Zachary J. Bell

Ms. Roy

CS 1401

Lab 5

Pseudo Code

**Algorithm lastSubStringMatch(Key, Target, index)**

1. lastIndex = -1
2. for (i = index; i <= target length – key length; ++i)
   1. charCount <- 0
   2. for (j = 0; j < key length; ++j)
      1. if (target character at (i + j) is equal to key character at j)
         1. ++charaCount
      2. Else break
   3. If charCount equals key length
      1. lastIndex = i + charCount – 1;
      2. break
3. give occurrences lastIndex

**Algorithm countSubStringMatch(Key, Target)**

1. For (i = 0; i < (length of Target – length of key); ++i)
   1. If (lastIndexSubStringMatch(Key, Target, i) != -1
      1. ++occurances
   2. i = lastIndexSubStringMatch(Key, Target, i);
2. Give occurances

**Algorithm ratioSubStringMatch(Key, Target)**

1. Ratio <- (( Key length \* call countSubStringMatch(Key, Target)) / Target length) \* 100.0
2. Give Ratio

**Algorithm SubStringMatch(filename, first, second)**

1. Set up scanner sc
2. For (i = 0; i < the max of first or second; ++i)
   1. If (the scanner has a next value in filename)
      1. If (first == second)
         1. FirstString <- sc nextline
         2. SecondString <- FirstString
      2. Else if ( first == (i + 1) )
         1. FirstString <- sc nextline
      3. Else if (second == (i + 1)
         1. SecondString <- sc nextline
      4. Else go to next line
   2. Else if (FirstString is empty or SecondString is empty) {ratio <- -1 }
3. If (FirstString length > SecondString length)
   1. Ratio <- call ratioSubStringMatch(SecondString, FirstString)
4. Else {ratio <- call ratioSubStringMatch(FirstString, SecondString)
5. Return ratio

**Algorithm MaxRatioSubString(filename) \*CHALLENGE\***

1. Set up scanner sc
2. While scanner has next line
   1. Go to nextline
   2. ++counter
3. For (i = 1; i <= counter; ++i)
   1. For (j = i + 1; j <= counter; ++j)
      1. If (call SubStringMatch(filename, i, j) > bestMatch)
         1. bestMatch <- call SubStringMatch(filename, i, j)
         2. first <- i
         3. second <-j
4. Write “Best match” + first + second + bestMatch to file
5. Give first, FirstString, second, SecondString