Jacob Alspaw
jaa134
EECS 340 - Algorithms
Assignment 4

Problem 1

```
SPARSE-TRANSPOSE(R, C, V, m, n, k) {
    R' ← empty array of n+1 elements
    C' ← empty array of length k
    V' ← empty array of length k
    //solve for R' in $\Theta$(k+n)
    R'[0] = 0 //R always starts with 0
    for (i ← 0 to k) {
        //increments the count of non-zero elements per row in A'
        R'[ C[i]+1 ] ← R'[ C[i]+1 ] + 1
    }
    for (i ← 1 to n+1) {
        //adds the previous values to the current value in R'
        R'[ i ] ← R'[i-1] + R'[i]
    }
    //solve for C' and V' in $\Theta$(k+m)
    j ← 0 //the location for the non-zero elements to put into C'
    D ← R' //copy R' into D expect for the last value of R'
    for (i ← 0 to k) {
        //puts the location of the i$^{th}$ non-zero value into C'
        C'[ D[C[i] ] = j
        //puts the value of the i$^{th}$ non-zero value into V'
        V[ D[C[i] ] ] ← V[i]
        //increments D to get the locations in C' and V'
        D[ C[i] ] ← D[ C[i] ] + 1
        //if you shifted rows
        if ( C[i+1] < C[i] ) {
            //then shift columns in A
            j ← j+1
        }
    }
}
```

FINAL RUN-TIME WILL BE Θ(m + n + k)

Problem 2

```
STABLE-SORT(A) {
    maxV ← max(A) //can find max element in linear time
    C ← empty array of length maxV
    //finding element frequencies
    for (i ← 0 to A.length) {
        C[A[i]] ← C[A[i]] + 1
    }
    //making the frequencies cumulative
    for (i ← 1 to C.length) {
        C[i] ← C[i] + C[i-1]
    }
    //translate array
    B ← empty array of length A.length
    for (i ← 0 to A.length) {
        B[i] ← A[i] + C[A[i]]
        C[A[i]] ← C[A[i]] + 1
    }
    MYSTERY-SORT(A)
    //transition back
    for (i ← 0 to A.length) {
        B[i] ← B[i] - C[A[i]]
        C[A[i]] ← C[A[i]] - 1
    }
    Return B
}
```