Solution:

Algorithm 1 SPARSE-TRANSPOSE(R, C, V, m, n, k)

```
1: R' ← new array[0...n] initializing with 0
2: C' ← new array[0...k−1] initializing with 0
3: V' ← new array[0...k−1] initializing with 0
4: T ← new array[0...n−1] initializing with 0
5: for i ← 0 to m−1 do
6:     for j ← R[i] to R[i+1]−1 do
7:         R'[C[j]] = R'[C[j]] + 1
8:     end for
9: end for
10: for i ← 1 to n do
11:     R'[i] = R'[i−1] + R'[i]
12: end for
13: for x ← 0 to m−1 do
14:     for y ← R[x] to R[x+1]−1 do
15:         temp = R'[C[y]] + T[C[y]]
16:         T[C[y]] = T[C[y]] + 1
17:         C'[temp] = x
18:         V'[temp] = V[y]
19:     end for
20: end for
21: return (R', C', V')
```

Explanation:

line 5 to line 12:

Line 5 to line 12 intends to construct R'. Instead of counting the number of nonzero entries in each row, R' records the number of nonzero entries in each column. This is because the transposition causes the rows in A to become the columns in A'. The outer for loop iterates from the first row of matrix A to the last row of matrix A. The inner for loop gets the column indexes of those nonzero entries in that row. Each time we encounter a nonzero entry, we increment the value with that column index by 1. In other words, every time we encounter a nonzero entry in the column of A, we know the transposed row will have 1 more nonzero entry.

line 10 to line 12 transfer the array into the culmulative array.

line 13 to line 20:

Line 13 to line 20 constructs C' and V'. The outer for loop iterates through the rows of matrix A (which is the same as looping through the columns of matrix A'). Its purpose is to find those entries whose nonzero entries are in column x (with respect to matrix A'). The inner for loop examines the nonzero entries in that row. For example, when $x = 0$, y iterates through $R[0]$ to $R[1] - 1$, which means it will examine the two nonzero entries in row 0. If we use these indexes to look up in C (e.g. $C[R[0]]$), we will find the column index of that element in A.

For each iteration of x, we are actually finding the positions of those nonzero entries whose column index is x. In line 15, $C[y]$ indicates the positions of the nonzero entries in row x (with respect to A), and since $R'[i]$ records the number of nonzero entries prior to the column i, it can tell us the index of the nonzero entry in C'. There will be two cases:

- It is the only nonzero entry in that column (in A), or it is the only nonzero entry in that row (in A'). In this case, we can safely use $R'[C[y]]$ as the index.
- There are other nonzero entries before it in this row (with respect to A'). In this case, we need the T array to keep track of the number of those entries in order to increment the index manually.

Therefore, line 17 find the position and value for $C'[temp]$ successfully. Similar task can be performed to find $V'[temp]$.