Read the compressed file into python as a list, need O(N) time where is the length of original text. Loop through the list delete all '\n' symbols and split every element of the list by space, need O(N) time. Overall need O(N) time and O(N) space to store the list.

Create two lists, loop through the list used before, append every string x times into two lists where x is the number before that string, need O(N) time. Use counting sort to sort one of the two lists, call first column, call another one last column. Overall, need O(N+M) time and O(M+N) space because counting sort need O (M + N) time and space.

Create rank list as an array. Because first column is sorted, go through the list, if the new item read is different from previous one, assign the index of that item into the array according the ASCII code of that item. Overall need O(N) to loop through first column and O (M) space to store the rank list.

Create a list call num_last which is used to support numbering last column, initialize all element to be 0. Go through the last column, for every item, go to the num_last array with index according to the ASCII code of the item, add the value by 1. Change the item to be (item, number of the that item in num_last). Overall need O(N) time to go through last column and O(M) space to store num_last.

Use reverse_bwt algorithm, initialize row to 0 and ori to '$', loop for length(bwt string) -1 times. Start from first row, get the string in the last column and prepend it to the ori. Go to the row where this string appears in the first row. Loop end. Replace '*' by space '-' by '\n' and delete '$'. Overall need O(N) time and space.