PRINCETON UNIVERSITY

COS 226–Algorithms and Data Structures

Week 10: *Radix Sorts, Suffix-Arrays & Compression (Algs. §5.1 & §5.1 & videos §19.A to F & 20.A to D)*

Version: November 29, 2017

Exercise 1 – Radix sorting

Put an X in each box if the string sorting algorithm (the standard version considered in class) has the corresponding property. When considering sublinear runtime, recall that n is the total number of characters in all strings (not the number of strings)

	mergesort	LSD radix sort	MSD radix sort	3-way radix quicksort
stable				
in-place				
sublinear time (in best case)				
fixed-length strings only				

Exercise 2 – Burrows-Wheeler Transform

The BurrowsWheeler compression algorithm consists of three algorithmic components, which are applied in succession:Burrows-Wheeler transform, move-to-front, and Huffman compression. In this exercise, we learn how to transform a given text using Burrows-Wheeler transform and use inverse Burrows-Wheeler transform to find the original text.

A. What is the Burrows-Wheeler transform of w h e e l e r?

suffix[0] = w h e e l e r
suffix[1] =
suffix[2] =
suffix[3] =
suffix[4] =
suffix[5] =
suffix[6] =

Sorted Suffixes
suffix[0] =
suffix[1] =
suffix[2] =
suffix[3] =
suffix[4] =
suffix[5] =
suffix[6] =

Write your answer :

B. Apply the Burrows-Wheeler inverse transform to find the original string

6
t[] = helweer
Construct the next array as shown in the Burrows-Wheeler assignment and find the original string. You only need to fill in the first character of sorted suffixes (in the box) and then complete the next array and find the original string.

i	Sorted suffixes	t	next
0	☐	h	
1	☐	e	
2	☐	l	
3	☐	w	
4	☐	e	
5	☐	e	
6	☐	r	

Exercise 3 – Compression(Bonus)

What is the compression ratio achieved by the following algorithms and inputs? Write the best-matching letter from the list provided (approximated ratios). For Huffman and LZW, assume that the input is a sequence of 8-bit characters (R = 256). Recall, the compression ratio is the number of bits in the compressed message divided by the number of bits in the original message.

A. 1/4096 B. 1/3840 C. 1/2560 D. 1/256 E. 1/255 F. 1/128 G. 1/32 H. 8/255 I. 1/16
J. 1/8 K. 1/7 L. 1/4 M. 1/2 N. 2/3 O. 1 P. 3/2 Q. 8

1. Run-length coding with 8-bit counts for best-case inputs of N bits.
2. Run-length coding with 8-bit counts for worst-case inputs of N bits.
3. Huffman coding for best-case inputs of N characters
4. Huffman coding for worst-case inputs of N characters.
5. LZW coding for best-case inputs of N characters using 12-bit codewords. Recall: no new codewords are added to the table if the table already has $2^{12} = 4096$ entries.
6. LZW coding for worst-case inputs of N characters using with 12-bit codewords. Recall: no new codewords are added to the table if the table already has $2^{12} = 4096$ entries.