University of Sciences, Faculty of information technology

Data Structures and Algorithms

Lab 3: Sorting and searching

You are given a file ***englishword.txt*** which contain name of people.

Part 1

Task 1:

Read ***englishword.txt*** file and generate four arrays from that.

1. a.txt (Random. Same ordering as the input file)
2. b.txt (Sorted. Can use any sorting algorithm or library to do sort in this part)
3. c.txt (Sorted in reverse order)
4. d.txt (Almost sorted. You need to perform (0.1 * length of array) number of swaps to perform this)

Task 2

Write the program to sort the above array in alphabetical order. The program takes two arguments:

- The first is the input file name. (a.txt, b.txt, c.txt, d.txt from ***Task 1***)
- The second is the sorting algorithm to perform
  - Bubble sort
  - Selection sort
  - Insertion sort
  - Merge sort
  - Quick sort
  - Radix sort

Print the name of the algorithm used, the array used (a, b, c, or d), running time and your ID.

Task 3

Do thoroughly experiment. Fill in this table run time of the function (in milliseconds)

	a	b	c	d
Bubble sort				
Selection sort				

Insertion sort				
Merge sort				
Quick sort				
Radix sort				

Make your analysis based on the above table. What is best in each case?

Part 2

Task 4:

From the file ***englishword.txt***, create 4 files:

1. 100.txt: randomly choose 100 names from names.txt, can be repeated.
2. 1k.txt: randomly choose 1k names from names.txt, can be repeated.
3. 10k.txt: randomly choose 10k names from names.txt, can be repeated.
4. 50k.txt: randomly choose 50k names from names.txt, can be repeated.

Task 5:

Make experiment with basic searching. You will search an item in file ***names.txt***

You will write search in 2 ways:

A. Linear search

B. Sort the array then applies Binary search (BS)

Fill in this table time run of the function (in milliseconds)

	100	1k	10k	50k
Linear Search				
Bubble sort + BS				
Selection sort + BS				
Insertion sort + BS				
Merge sort + BS				
Quick sort + BS				
Radix sort + BS				

For example:

- Linear Search – 100: you do Linear Search 100 times, with 100 items from 100.txt file created in Task 4
- Merge sort + BS – 50k: you do Merge sort (1 time), then Binary Search 50,000 times with 50,000 items from 50k.txt file created in Task 4

Make your analysis based on the above table. What is best in each case?

Terms of submission

- Student are required to submit both source code, document and some additional files for this Lab.
- Compress them with the name <StudentID>.zip or <StudentID>.rar. Then submit this compressed file.

Similar source code, plagiarism or spam submissions will score 0 in this SUBJECT