

Лабораторная работа № 1 по курсу дискретного анализа: сортировка за линейное время

Выполнил студент группы 08-207 МАИ *Боев Савелий*.

Условие:

1. Требуется разработать программу, осуществляющую ввод пар «ключ-значение», их упорядочивание по возрастанию ключа указанным алгоритмом сортировки за линейное время и вывод отсортированной последовательности.

2. Вариант задания: 4.2

Тип сортировки: поразрядная сортировка.

Тип ключа: даты в формате DD.MM.YYYY, например 1.1.1, 1.9.2009, 01.09.2009, 31.12.2009.

Тип значения: строки переменной длины (до 2048 символов).

Метод решения

Структура **"item"** используется для хранения пары "ключ-значение". Тут храним трехэлементный массив целых чисел для хранения даты в формате DD.MM.YYYY и храним индекс элемента в исходном массиве.

Функция **Sort** принимает массив элементов типа **"item"**, его длину и позицию разряда для сортировки. В данной реализации ключом для сортировки является каждый отдельный разряд даты (день, месяц, год). Внутри функции создается вспомогательный массив **res** для хранения отсортированных элементов. Сначала вычисляется кол-во элементов с определенным значением разряда с помощью массива **count**. Затем происходит преобразование массива **count** так, чтобы каждый элемент содержал количество элементов, меньших или равных данному значению. После этого происходит перенос элементов во временный массив **res** в соответствии с их позицией, определенной по значению разряда исходного массива. Наконец,

отсортированные элементы из массива **res** копируются обратно в исходный массив **array**.

Функция **RadixSort** принимает массив элементов типа **"item"** и длину. Вызывается функция **Sort** для каждого разряда даты в порядке от младшего к старшему. Сортировка по дню, месяцу и году.

В функции **main** происходит чтение пар "ключ-значение" из стандартного ввода. Каждая пара вводится в формате "ключ\значение", где ключ представляет собой дату в формате DD.MM.YYYY, а значение - произвольная строка. Ввод прекращается при встрече символа новой строки. В процессе чтения происходит заполнение массива **v** структурами **"item"** и сохранение символов в массив **data**.

После завершения чтения вызывается функция **RadixSort** для сортировки массива **v** по дате.

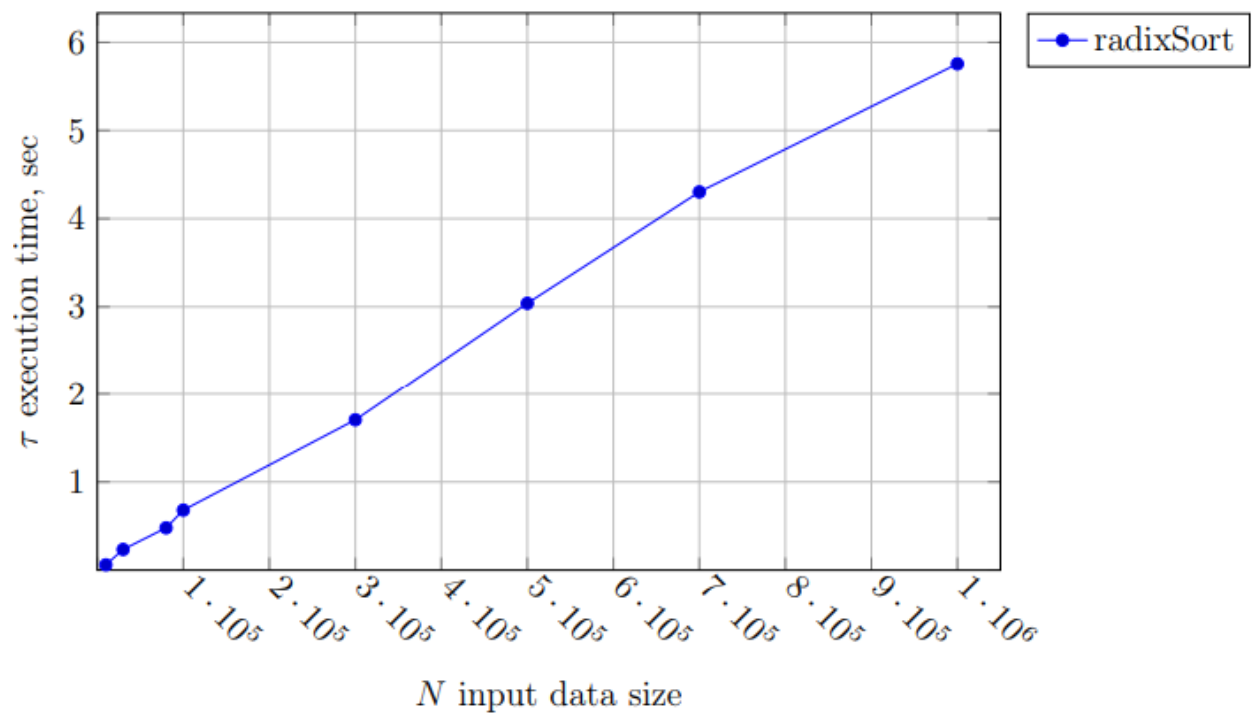
Вывод отсортированных пар "ключ-значение" в стандартный вывод. Индексы элементов, сохраненные в поле **ind** каждого элемента, используются для определения порядка вывода из массива **data**.

Наконец, освобождаются выделенные динамические массивы **data** и **v**, и программа завершается.

Описание программы

Программа состоит из одного файла. В ней присутствуют структуры вектор **char** (храним символы), вектор строк (храним пары ключ-значение), структура **item** (храним трехэлементный массив + индекс элемента). Для каждого вектора описаны функции создания этих векторов и добавления в эти вектора. И описаны функции для поразрядной сортировки и сортировки подсчетом.

Тест производительности



Линейность сортировки сохраняется.

Выводы

Данный алгоритм применен только к определенному формату ключа, то есть к формату дат. Так как я сохраняю всю строку, то можно сортировать любые строки с ключом-датой и любым значением. Кроме того, реализуемую функцию сортировки поразрядно и подсчетом можно использовать для сортировки чисел. Сложность программирования данной работы не очень высока, необходимо знать, как работают сортировки и быть внимательным к формату вывода и к реализуемым структурам.