

Санкт-Петербургский государственный университет

Доклад по предмету "Алгоритмы и анализ сложности"

LSD-поразрядная сортировка

Выполнил бакалавр 3 курса:
Полоз Алексей Евгеньевич

Санкт-Петербург
2018

Содержание

Описание, история разработки, область применения	3
Математический анализ	3
Характеристика данных, трудоёмкость	3
Реализация программы	3
Входные данные	5
Измерение	5
Используемые источники	5

Описание, история разработки, область применения

Поразрядная сортировка Radix Sort изначально была предназначен для сортировки целых чисел, записанных цифрами. Но так как в памяти компьютеров любая информация записывается целыми числами, алгоритм пригоден для сортировки любых объектов, запись которых можно поделить на «разряды», содержащие сравнимые значения.

Данный алгоритм не использует сравнений сортируемых элементов.

Есть два вида такой сортировки — MSD (most significant digit — сначала старший разряд) и LSD (least significant digit — сначала младший разряд). LSD несколько удобнее для сортировки чисел, т.к. не приходится «приписывать» к числам слева незначащие 0 для выравнивания числа разрядов. MSD же удобнее для сортировки строк.

Математический анализ

Сравнение производится поразрядно: сначала сравниваются значения одного крайнего разряда. Сортировка в таком случае выполняется с помощью сортировки подсчетом (count sort). Сложность — $O(n)$. Далее элементы группируются по результатам этого сравнения, затем сравниваются значения следующего разряда, соседнего, и элементы либо упорядочиваются по результатам сравнения значений этого разряда внутри образованных на предыдущем проходе групп, либо переупорядочиваются в целом, но сохраняя относительный порядок, достигнутый при предыдущей сортировке. Затем аналогично делается для следующего разряда, и так до конца.

Характеристика данных, трудоёмкость

До сортировки необходимо знать два параметра: количество разрядов в самом длинном ключе и разрядность данных — количество возможных значений разряда ключа.

Шагов столько, сколько разрядов в числах. Соответственно, сложность алгоритма — $O(n*k)$, k — число разрядов.

Алгоритму в такой реализации требуется дополнительная память — $O(n)$.

Реализация программы

Пример реализации в личном репозитории на C++:

<https://github.com/kosyachniy/dev/blob/master/algorithm/sort/Radix%20Sort%20LSD.cpp>

Пример реализации в личном репозитории на Python:

<https://github.com/kosyachniy/dev/blob/master/algorithm/sort/Radix%20Sort%20LSD.py>

```
1  from math import log
2
3
4  def getDigit(num, base, digit_num):
5      return (num // base ** digit_num) % base
6
7  def makeBlanks(size):
8      return [ [] for i in range(size) ]
9
10 def split(a_list, base, digit_num):
11     buckets = makeBlanks(base)
12     for num in a_list:
13         buckets[getDigit(num, base, digit_num)].append(num)
14
15     return buckets
16
17 def merge(a_list):
18     new_list = []
19     for sublist in a_list:
20         new_list.extend(sublist)
21
22     return new_list
23
24 def maxAbs(a_list):
25     return max(abs(num) for num in a_list)
26
27 def split_by_sign(a_list):
28     buckets = [ [], [] ]
29     for num in a_list:
30         if num < 0:
31             buckets[0].append(num)
32         else:
33             buckets[1].append(num)
34
35     return buckets
36
37 def radixSort(a_list, base):
38     passes = int(round(log(maxAbs(a_list), base)) + 1)
39     new_list = list(a_list)
40     for digit_num in range(passes):
41         new_list = merge(split(new_list, base, digit_num))
42
43     return merge(split_by_sign(new_list))
44
45
```

Входные данные

Входные данные в конкретной реализации представляют из себя массив десятичных чисел.

Для чистоты эксперимента можно брать числа разной длины, повторяющейся длины, повторяющиеся числа, с повторяющимися цифрами в разрядах.

Генерация данных:

```
1 import numpy as np
2 import random
3
4 print(np.array([random.random() for i in range(10)]))
```

Измерение

Выполняется за линейное время $O(nk)$.

Используемые источники

- <https://habr.com/post/268261/>
- http://algolist.manual.ru/sort/radix_sort.php
- [https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%80%D0%B0%D0%B7%D1%80%D1%8F%D0%B4%D0%BD%D0%B0%D1%8F_%D1%81%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0#%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_\(LSD\)](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%80%D0%B0%D0%B7%D1%80%D1%8F%D0%B4%D0%BD%D0%B0%D1%8F_%D1%81%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0#%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_(LSD))