



ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
**ВЫСШАЯ ШКОЛА ЭКОНОМИКИ**

## Учебная практика 2017

# Сортировки

## Линейные сортировки

Сортировка подсчетом

Цифровая сортировка



# Сортировка подсчетом

## Counting Sort

Каждый из  $n$  элементов сортируемого массива – целое положительное число

$$0 < A[i] \leq k, \quad k \text{ заранее известно}$$

Результат записывается в другой массив того же размера

Используется вспомогательный массив размером  $k+1$  элементов

**Сортировка устойчивая – сохраняется взаимное расположение одинаковых элементов**

Исходный массив

5 4 3 5 2 1 4

Сортированный массив

1 2 3 4 4 5 5



# Пример

На олимпиаду пришли 507 школьников из 50 школ района.

Информация на вход – ФИО школьника, номер школы.

Определить по три школы, количество участников из которых максимально и минимально.

Вывести номера школ и количество участников по убыванию.

Решение?



# Упражнение

Задан массив

i	0	1	2	3	4
A	3	2	3	4	1

Вычислить значение

$$B = A[A[A[A[1] + 1] - A[0]] - A[4]]$$



# Сортировка подсчетом

```
StableCountingSort (A,B,k)
```

```
for i = 0 to k
```

```
    C[i] = 0
```

```
for i = 0 to n - 1
```

```
    C[A[i]] = C[A[i]] + 1
```

```
for j = 1 to k
```

```
    C[j] = C[j] + C[j - 1]
```

```
for i = n - 1 to 0
```

```
    B[C[A[i]]-1] = A[i];
```

```
    C[A[i]] = C[A[i]] - 1
```

Обнуляем массив C

В C[i] – количество элементов массива A, равных i

Находим частичные суммы – количество элементов не больше i

Сортированный массив – в массиве B

Уменьшаем C[i] на 1, если в массиве A есть повторения, они запишутся в свою позицию



# Сортировка подсчетом

A

Пример

0	1	2	3	4	5	6	7
3	6	4	1	3	4	1	4

C

0	1	2	3	4	5	6
0	2	0	2	3	0	1

C

0	1	2	3	4	5	6
0	2	2	4	7	7	8

for i = 0 to n - 1

$C[A[i]] = C[A[i]] + 1$

B

0	1	2	3	4	5	6	7
						4	

C

0	1	2	3	4	5	6
0	2	2	4	6	7	8

for j = 1 to k

$C[j] = C[j] + C[j-1]$

i=6

$B[C[A[6]]-1] = B[C[1]-1] = B[1] = A[6]$

B

0	1	2	3	4	5	6	7
	1					4	

C

0	1	2	3	4	5	6
0	1	2	4	6	7	8

for i = n - 1 to 0

$B[C[A[i]]-1] = A[i];$

$C[A[i]] = C[A[i]] - 1$

i=7  $B[C[A[7]]-1] = B[C[4]-1] = B[6] = A[7]$



# ЦИФРОВАЯ СОРТИРОВКА



# Цифровая сортировка (radix sort)

Сортировка проводится по разрядам, причем начиная с младшего.

Важно сохранять порядок расположения чисел таким же, каким он был на входе, при совпадении значений разряда

Сортировку по значению разряда можно провести подсчетом

Важно правильно выбрать основание системы счисления

Если основание СС учитывает формат хранения данных (двоичная СС, байты, 32-битные целые), можно уменьшить расходы на выделение цифр.

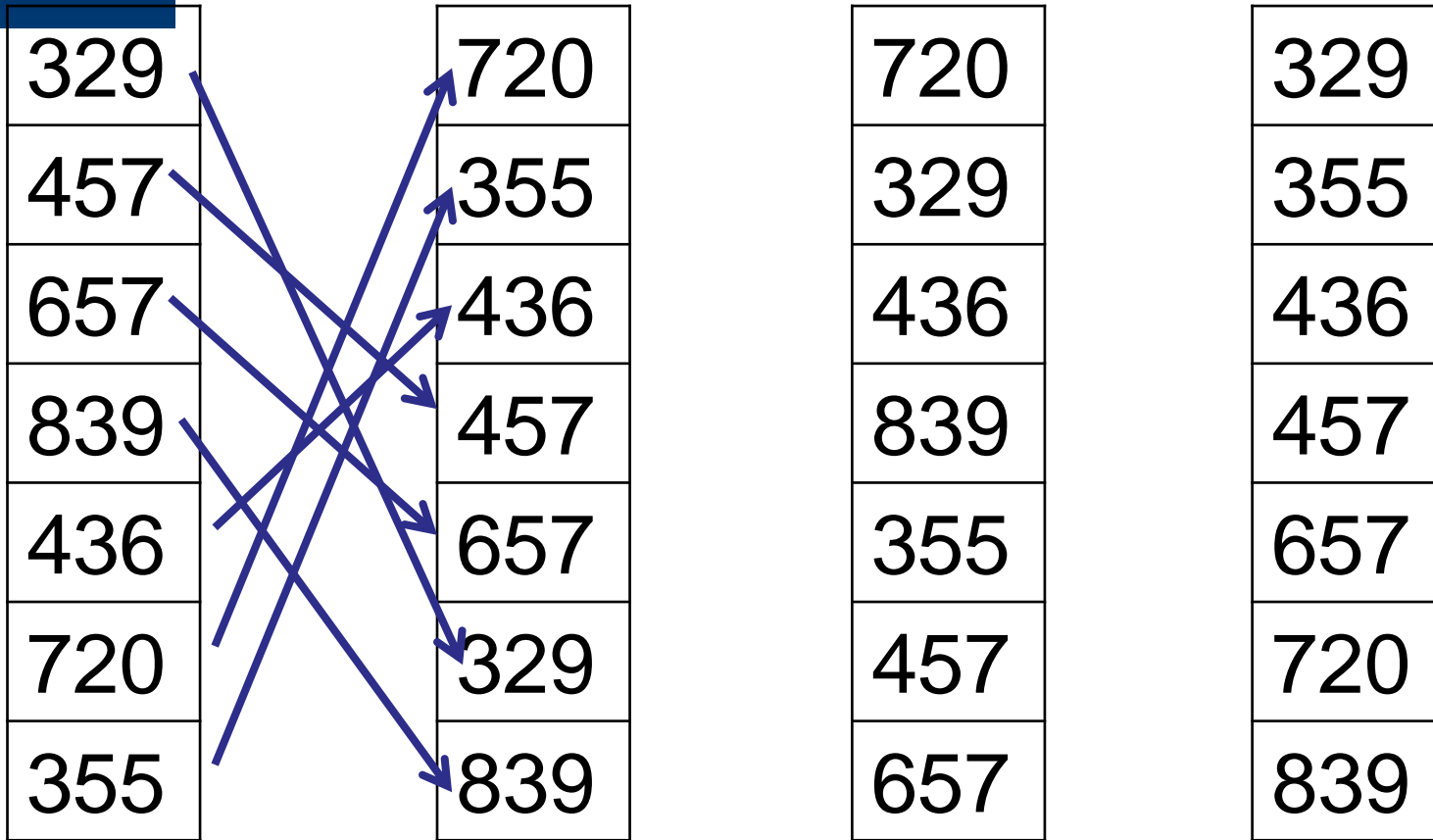
Задача – выделить цифры числа по основанию 256.

Сравнивая отдельные цифры, не забывайте о том, что обменивать надо числа.





# Цифровая сортировка (radix sort)



Важно сохранять порядок расположения чисел при совпадении значений разряда



# Выводы

Сортировка подсчетом – устойчивая, линейной сложности.

Линейные (псевдолинейные) алгоритмы сортировки

- имеют ограничения – массив должен состоять из целых чисел, не вещественных.
- Если диапазон значений массива большой, возрастают расходы памяти на вспомогательные массивы

Цифровая сортировка может эффективно применяться для целых без знака. Если использовать `union`, не будет больших затрат на выделение цифр числа

