их упорядочим (естественно, $0 < 1 < \ldots < k-1$) и присвоим соответствующим вагонам.

(2) Алгоритм начинается с нулевого шага.

(3) На l-м шаге распускаем состав в соответствии с символами в l-м разряде, как в поразрядной сортировке.

(4) Затем вытягиваем вагоны из 0-го стека (т.е. того стека, где собрались вагоны, у которых в l-м разряде стоит 0).

(5) Перенумеровываем стеки для $(l+1)$-го шага в соответствии с символами в $(l+1)$-м разряде, пустой стек получает номер $k-1$.

Следствие 3. *Число $(p-1)$ буквенных слов в языке, порожденном грамматикой Q_k, равно $u_k(p)$.*

Следствие 4. *Если $p \le k$, то алгоритм (K) совпадает с алгоритмом (A).*

Следствие 5. *Если $p \le k$, то число $(p-1)$-буквенных слов в языке, порожденном грамматикой Q_k, равно $u_k(p) = 2^{p-1}$.*

В одной из первых задач спрашивалось, как бы вы предложили нумеровать вагоны, если получателю неважно, в каком порядке он получит свои вагоны?

Обычно все вагоны для одного получателя занумеровывают одним номером. Такой объект, где есть несколько первых номеров, несколько вторых, ..., несколько n-х называют *мультимножеством*. Можно рассматривать перестановки мультимножества

$$S = \{\underbrace{1\ldots1}_{m_1}\underbrace{2\ldots2}_{m_2}\ldots\underbrace{s\ldots s}_{m_s}\}.$$

Задача 20. Введите для мультимножества понятие блока.

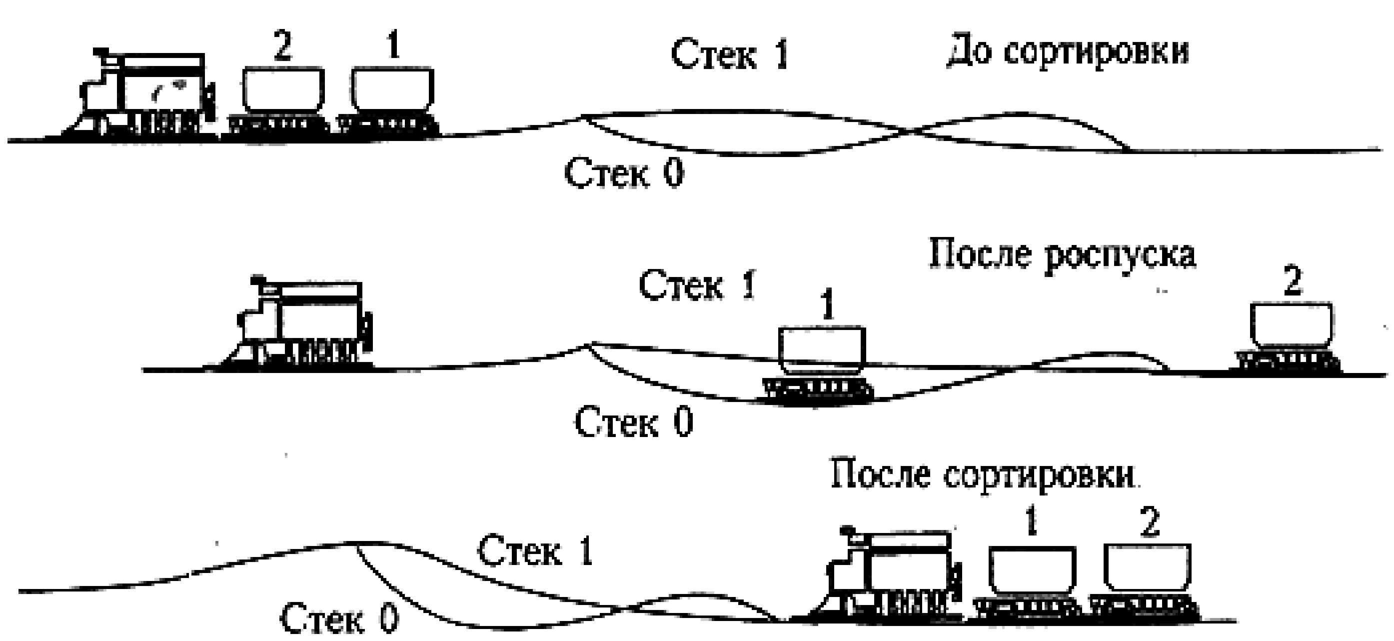

Рис. 9

Алгоритм (K) вместе с приведенными выше результатами полностью завершает решение задачи о **минимальном числе вытягиваний при сортировке железнодорожных составов.**

Заключение

Итак, задача решена полностью. Теперь хочется остановиться и оглядеться. Давайте посмотрим, что мы получили. Основное — это таблица $W(p, k)$. Если вы в детстве считали вагоны проходящих мимо вас составов, то могли бы заметить, что редко их бывает больше 60, а число получателей редко бывает больше 20. Рассмотрев окончательную таблицу чисел $W(p, k)$ (рис.8), можно увидеть, что разумно делать горки с 3–4 стеками, но даже горка с 2 стеками не сильно увеличивает число вытягиваний. Но повысит ли наш алгоритм существенно производительность? Ведь когда состав сортируется на горке, она занята для других составов и они простаивают. А нельзя ли интенсифицировать процесс сортировки как-нибудь по-другому? Что мешает сортировать сразу несколько составов? Это принцип устройства горки — в стек можно класть с одной стороны и вынимать из него тоже с той же стороны. А не изменить ли устройство горки, сделав ее похожей на «очередь», в которую с одной стороны можно «класть» вагоны, а «вынимать» можно с другой? Но большинство составов не отсортируешь за один шаг. А не сделать ли несколько горок, соединенных последовательно, да еще и замкнуть всю эту систему в кольцо! Как, например, на рисунках 9 и 10.

Правда, такое кольцо должно занимать много места, гораздо больше, чем нынешние горки. И вот тут возникает последняя идея: а не убрать ли все это под землю! Ведь существование окончательного алгоритма означает, что составы можно сортировать без какого-либо присутствия человека. А вместо железнодорожных путей можно будет разбить прекрасные парки.

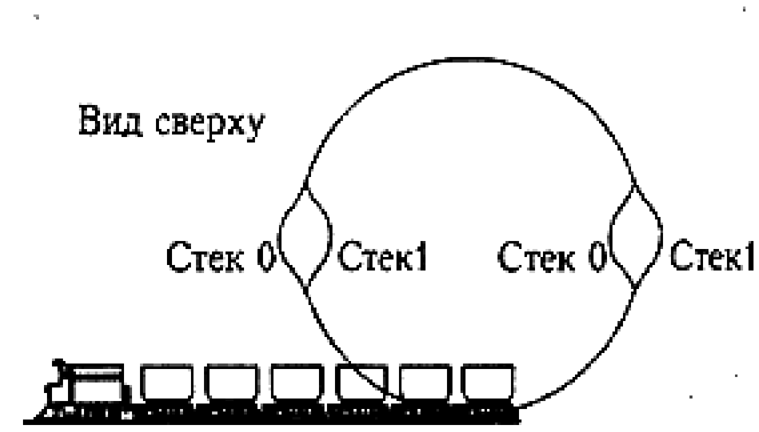

Рис. 10

Задачи для самостоятельного исследования

Задача 1*. Оцените, каков порядок числа вагонов W, прошедших через стрелку S, в алгоритмах, приведенных в этой статье.

Задача 2*. Придумайте алгоритм сортировки, который каждую индивидуальную перестановку сортирует за наименьшее число

а) шагов,

б) вытягиваний,

если в каждом стеке помещается ограниченное число вагонов $n_1, n_2, \ldots, n_k$.

Задача 3.** Пусть теперь разрешается распускать не весь состав и вытягивать из стеков не обязательно всю группу вагонов. Придумайте алгоритм сортировки, в котором каждая индивидуальная перестановка сортируется за наименьшее число W вагонов, прошедших через стрелку S.