АиСД_дз5

CovarianceMomentum

1term

Во всех задачах я буду пользоваться восхитительным трюком с заменой чисел на нули и единицы и последующим приведением одного к другому через монотонную функцию.

1

Пойдём от противного. Пусть всего в сортирующей сети n нитей. Рассмотрим ситуацию, когда на входах сначала $i - 1$ ноль, затем единица, затем ноль, оставшиеся чила — нули. Получаем что-то вида 00000010111111. Таким образом, ни одна из единиц, находящихся на позициях $[i+2; n]$ не может встать на позицию $i + 1$, так как им для этого придётся "подняться по сети а единица с позиции i не может встать на неё, так как её нить никогда не сравнивается с нитью $i + 1$, а подняться выше она не может.

2

Заметим, что количество позиций, на которые может встать новый элемент среди других $n - 1$, равно n — перед каждым из них и после всех. Каждый компаратор возвращает один из двух результатов. Таким образом, если при любых результатах действий компараторов наш элемент проходит $f < \log_2 n$ компараторов, то всего есть не более $2^f < n$ вариантов его конечного расположение, противоречие.

3

Давайте построим один слой из n компараторов, сравнивающих каждый i-ую и $2n - i + 1$-ую нити. Пусть единиц в первом массиве o_1, во втором o_2. Поймём, что после такого слоя во втором массиве окажется $min(n, o_1 + o_2)$ единиц. Заметим, что компараторы вида $(i, 2n - i + 1)$, где $2n - i + 1 \geq 2n - o_2$, то есть $o_2 + 1 \geq i$, ничего не испортят во втором массиве, и на нижних o_2 позициях будут стоять единицы. Теперь заметим, что на позициях $[n+1; n+o_1]$ тоже будут стоять единицы, так как компараторы перенесут сюда единицы из первого массива. Таким образом, во втором массиве будет всего $o_1 + o_2$ единиц. Ну иногда такое случается, что это дело не влезет, тогда там будет $2n$. Меньше не может быть, потому что отрезки единиц в результирующем втором массиве идут с разных сторон и перескаются только если весь второй массив уже заполнен.

4

Будем говорить, что по компараторам единичка "спускается а по антикомпараторам — "поднимается". Также во время рассуждений будем считать, что если на концах компаратора две нити с одинаковыми значаениями, то интересующая нас единица будет подниматься по антикомпаратору и спускаться по компаратору всё равно.

Представим, что на каждом входе стоит единица, и "пройдём"по её пути. Каждый раз, когда мы будем встречать компаратор, по которому можно спуститься или антикомпаратор, по которому можно подняться, будем переходить по нему. Рассмотрим конкретно путь по антикомпаратору вверх. Когда встречаем такой путь на нити i, продолжаем следовать вверх по нему до тех пор, пока не придём на нить j, которая находится ниже этой. Тогда создадим компаратор $(i;j)$ в нашей сети. Сделаем так со всеми единицами, а затем выкинем все антикомпараторы. Поймём, что мы создали не более c_2 антикомпараторов — потому что для каждого антикомпаратора однозначно определён путь, по которому мы пошли, значит, однозначно определён создаваемый компаратор. Поймём, что мы создали не менее c_2 компараторов — потому что каждый антикомпаратор задействован хотя бы в одном пути, а если после прохода вверх по антикомпаратору мы не спустились обратно, то наша сеть — не сортирующая. Поймём, что полученная сеть — сортирующая. Это верно потому, что для любой единицы мы "укоротили"её путь до нужной нити.

5