Лекция №3

Условие связывания блоков отношением упорядочивания

$A_k < A_j$
$\forall a_{kn`+l} \in A_k,\ \forall a_{jn`+s} \in A_j$ *(1)*
$a_{kn`+l} \geq a_{jn`+s} \Rightarrow A_k \geq A_j$
$l = 0..(n`-1),\ s = 0..(n`-1)$

$A_k < A_j$
$\forall a_{kn`+l} \in A_k,\ \forall a_{jn`+s} \in A_j$ *(2)*
$a_{kn`+l} < a_{jn`+s} \Rightarrow A_k < A_j$
$l = 0..(n`-1),\ s = 0..(n`-1)$

В случае если элементы блоков A_k и A_j являются упорядоченными, тогда условие *(1)* будет проинтерпретированно следующим образом:
$a_{kn`} \geq a_{(j+1)n`-1} \Rightarrow A_k \geq A_j$ – первый элемент блока A_k не меньше последнего элемента блока A_j

Выражение (2) может быть проинтерпретированно следующим образом:
$a_{(k+1)n`-1} < a_{jn`} \Rightarrow A_k \geq A_j$ – первый элемент блока A_k не меньше последнего элемента блока A_j

Алгоритм блочной «чёт-нечётной» перестановки

1. Для пар блоков вида (A_0, A_1), (A_2, A_3), … проверяется выполнение условия $A_{2j} <= A_{2j+1}$ *(3)* Если данное условие не выполняется, тогда реализуется упорядочивание элементов в паре блоков с использованием алгоритма сортировки слиянием. Иначе никаких действий с парой блоков не выполняется.
2. Для пар блоков вида (A_1, A_2), (A_3, A_4), … проверяется выполнение условия $A_{2j-1} <= A_{2j}$ *(4)* Если данное условие не выполняется, тогда реализуется упорядочивание элементов в паре блоков с использованием алгоритма сортировки слиянием. Иначе никаких действий с парой блоков не выполняется.

Распараллеливание блочного варианта «чёт-нечётной» перестановки предполагает возможность одновременной проверки условий *(3)* и *(4)* и в случае их не выполнения для определённых пар блоков выполняется сортировка слиянием с последующим разделением блоков.

Сортировка Шелла

Последовательный вариант сортировки

Особенность реализации перестановки – это обмен при выполнении условия сортировки между элементами находящимися друг от друга на различных «расстояниях».

На первом этапе реализуется рассмотрение групп элементов по два элемента, на втором по 4, на третьем по 8 и т.д. На заключительном этапе группой является весь массив.

Правило (способ) формирования групп элементов на соответствующих этапах:

1) $(a_i, a_{n/2+i})$ $i = 1..(n/2)$
2) $(a_i, a_{n/4+i}, a_{2n/4+1}, a_{3n/4+1})$ $i = 1..(n/4)$
3) $(a_i, a_{n/8+i}, a_{2n/8+1}, a_{3n/8+1}, a_{4n/8+i}, a_{5n/8+i}, a_{6n/8+i}, a_{7n/8+i})$ $i = 1..(n/8)$

На каждом этапе количество элементов в группе увеличивается на 2. Внутри рассматриваемой группы на каждом этапе производится сортировка.

Пример реализации последовательной сортировки алгоритмом Шелла

1. 16 7 10 1 13 11 3 8 14 4 2 12 6 5 9
2. САААААМИ
3. к экзамену

Параллельная реализация сортировки Шелла

<u>Гиперкуб</u> – это топология с соединением узлов кластера, а также это топология взаимодействия между параллельно-выполняющимися процессами.

Обозначим через N размерность гиперкуба, как топологии кластера, тогда при N = 1 количество узлов в кластере $P = 2^N = 2$

Предполагается, что n (количество элементов в исходном массиве) равно 16. Для реализации сортировки количество процессов определяется как 2P = 4 (при N = 1).

Идентифицируем номера процессов как 00, 01, 10, 11 (N+1) – размерность гиперкуба для процессов.

Правило определения номеров взаимодействующих процессов (номеров блоков данных, которые будут передаваться между процессами). Если i номер итерации (i = 0..N), то на i-ой итерации обмен выполняют те процессы, у которых различие в битовом представлении их номеров имеется в разряде N-i, при этом разряды сравниваются слева направо. При N = 1, i = 0 для первых пар блоков различие должно быть в 1-ом разряде. При N = 1 для вторых пар блоков различие в 0-ом разряде.

Таким образом при N = 1 должно быть выполнено 2 итерации на которых реализуется обмен блоками между процессами с соответствующими номерами и выполнение этими процессами операции сравнить и разделить.