8.3. Stripe Approach Illustration

Let's say,

mapper-0 input

24, 31, 7, 81, 31

mapper-1 input

7, 20, 81

24 % 4 = 0
31 % 4 = 3
7 % 4 = 3
81 % 4 = 1
20 % 4 = 0

In Stripe approach, we maintain an associative array (eg. hashtable) for neighbors for each term. So, for mapper-0, the output will be as follows:

(24, {31:2, 7:1, 81:1}); (31, {7:1, 81:1}); (7, {31:1, 31:2})

(81, {31:1})

Similarly, for mapper-1,

(7, {20:1, 81:1}); (20, {81:1})

Now, the mapper outputs are shuffled by using some partition logic and then sorted and grouped making ready for input for reducers. The I/O for each reducer is given below:

use: partition = (int) key % r

R0

Input: (20, [81:1])
(24, [31:2, 7:1, 81:1])

output: (20, {81:1})
(24, {31:2, 7:1, 81:1})

R1

(81, [31:1])

(81, {31:1})

R2

R3

(7, [20:1, 81:1, 81:1, 31:1])
(31, [7:1, 81:1])

(7, {20:1, 81:2, 31:1})
(31, {7:1, 81:1})