**02-线性结构3 Reversing Linked List (25 分)**

Given a constant *K* and a singly linked list *L*, you are supposed to reverse the links of every *K*elements on *L*. For example, given *L* being 1→2→3→4→5→6, if *K*=3, then you must output 3→2→1→6→5→4; if *K*=4, you must output 4→3→2→1→5→6.

Input Specification:

Each input file contains one test case. For each case, the first line contains the address of the first node, a positive *N* (≤10​5​​) which is the total number of nodes, and a positive *K* (≤*N*) which is the length of the sublist to be reversed. The address of a node is a 5-digit nonnegative integer, and NULL is represented by -1.

Then *N* lines follow, each describes a node in the format:

Address Data Next

where Address is the position of the node, Data is an integer, and Next is the position of the next node.

Output Specification:

For each case, output the resulting ordered linked list. Each node occupies a line, and is printed in the same format as in the input.

Sample Input:

00100 6 4

00000 4 99999

00100 1 12309

68237 6 -1

33218 3 00000

99999 5 68237

12309 2 33218

Sample Output:

00000 4 33218

33218 3 12309

12309 2 00100

00100 1 99999

99999 5 68237

68237 6 -1