Leetcode Problem - 3 (725)
(Split linked list is parts)

```c
int getLength (struct ListNode* head) {
    int length = 0;
    while (head != NULL) {
        length++;
        head = head → next;
    }
    return length; }

struct ListNode ** splitListToParts (struct ListNode* head, int k, int *returnSize) {
    int length = getLength (head);
    int partSize = length / k;
    int remainder = length % k;
    struct ListNode ** result = (struct ListNode **) malloc (k * sizeof (struct ListNode*));
    *returnSize = k;
    for (int i = 0; i < k; i++) {
        int currentPartSize = partSize + (i < remainder ? 1 : 0);
        if (currentPartSize == 0)
            result [i] = NULL;
        else {
            result [i] = head;
            for (int j = 0; j < currentPartSize - 1; j++) {
                head = head → next;
            }
            struct ListNode *temp = head → next;
            head → next = NULL;
            head = temp;
        }
    }
    return result;
}
```

Output: head = (1, 2, 3)

K = 5

[[1], [2], [3], [ ], [ ]]