

```

/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     struct ListNode *next;
 * };
 */

typedef struct ListNode* node;

struct ListNode** splitListToParts(struct ListNode* head, int k, int* returnSize) {

    int cnt = 0;

    node temp = head;

    while (temp != NULL) {

        temp = temp->next;

        cnt++;

    }

    int part = cnt / k;

    int rem = cnt % k;

    temp = head;

```

```

node* arr = (node*)malloc(sizeof(node) * k);

for (int i = 0; i < k; i++) {

    node sub = NULL;

    node st = sub;

    for (int j = 0; j < (rem > 0 ? part+1 : part); j++) {

        if (sub == NULL) {

            sub = (node)malloc(sizeof(struct ListNode));

            st = sub;

        } else {

            st->next = (node)malloc(sizeof(struct ListNode));

            st = st->next;

        }

        st->val = temp->val;

        st->next = NULL;

        temp = temp->next;

    }

    arr[i] = sub;

    if (rem > 0) {

        rem--;

    }
}

```

```
}
```

```
*returnSize = k;
```

```
return arr;
```

```
}
```

OUTPUT:

**Accepted**  
sharan\_060704 submitted at Feb 05, 2024 11:36

[Editorial](#) [Solution](#)

**Runtime**  
3 ms  
Beats 70.00% of users with C

**Memory**  
6.70 MB  
Beats 60.00% of users with C

Runtime (ms)	Percentage of Users
1ms	~22%
2ms	~5%
3ms	~15% (User's position)
4ms	~5%
5ms	~5%
6ms	~30%
7ms	~5%
8ms	~2%

**Code | C**

```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     struct ListNode *next;
 * };
 */
```

[View more](#)

**More challenges**

- 61. Rotate List
- 328. Odd Even Linked List
- 2674. Split a Circular Linked List

