

Even-Odd linked list

19/11/24

LeetCode - 3

__/_/_

```
#include <stdio.h>
#include <stdlib.h>
```

```
struct listnode {
```

```
    int val;
```

```
    struct listnode *next;
```

```
};
```

```
struct listnode* oddEvenList ( struct listnode* head)
```

```
{
```

```
    if (head == NULL || head->next == NULL ||
```

```
        head->next->next == NULL)
```

```
        return head;
```

```
    struct listnode* odd = head;
```

```
    struct listnode* even = head->next;
```

```
    struct listnode* evenhead = even;
```

```
    while (even != NULL && even->next != NULL)
```

```
    {
```

```
        odd->next = even->next;
```

```
        even->next = odd->next;
```

```
        even = even->next;
```

```
    }
```


odd \rightarrow next = evenhead;

return head;

}

struct listnode* newnode(int val)

{

struct listnode* node =

(struct listnode*) malloc (size of
(struct listnode));

node \rightarrow val = val;

node \rightarrow next = NULL;

return node;

}

void printlist(struct listnode* head)

{

while (head != NULL)

{

printf("%d", head \rightarrow val);

head = head \rightarrow next;

}

printf("\n");

}

Case 1

Case 2

Input

Input

head =

head =

[1, 2, 3, 4, 5]

[2, 1, 3, 5, 6, 4, 7]

output =

output

[1, 3, 5, 2, 4]

[2, 3, 6, 7, 1, 5, 4]

expected

expected

[1, 3, 5, 2, 4]

[2, 3, 6, 7, 1, 5, 4]