

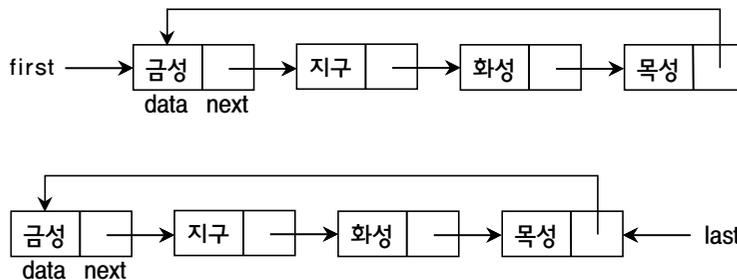
● 원형연결리스트(circular linked list)

원형연결리스트는 마지막 노드의 포인터가 첫 번째 노드를 가리키는 구조이다.
원형연결리스트의 헤드노드는 처음 또는 마지막을 가리킬 수 있다.

원형연결리스트는 다음 2가지 형태의 구조가 있다.

```

struct node
{
    char data[5];
    struct node *next; //next는 다음 노드를 가리키는 포인터
};
    
```



- 원형연결리스트는 모든 포인터가 다음 노드를 가리키고 있다.
- 원형연결리스트에서는 반드시 처음 노드부터 검색을 시작할 필요는 없다.
- 원형연결리스트에서 검색시 주의할 점은 무한루프에 빠지지 않게 해야 한다.
- 원형연결리스트에서 헤드노드를 이용하면 무한루프를 방지할 수 있다.
- 원형연결리스트는 사이클(cycle) 구조의 자료를 구현할 때 유용하다.

◆ 원형연결리스트에서

- 헤드가 처음보다는 **마지막** 노드를 가리키는 구조가 더 **우수한** 구조이다.
- 즉, first보다는 last가 더 편리하고 유용한 구조

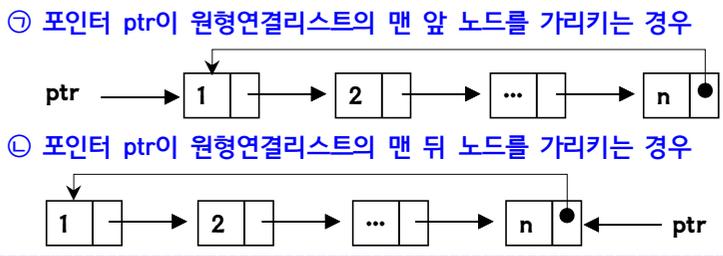
// 원형연결리스트에서 노드 삽입/삭제 연산시간

		리스트 처음 위치	리스트 마지막 위치
head가 첫번째 노드를 가리키는 구조	노드 삽입	$O(n)$	$O(n)$
	노드 삭제	$O(n)$	$O(n)$
head가 마지막 노드를 가리키는 구조	노드 삽입	$O(1)$	$O(1)$
	노드 삭제	$O(1)$	$O(n)$

- ① head가 리스트의 첫번째 노드를 가리키는 원형연결리스트에서 삽입/삭제
 - 모두 리스트 처음 노드부터 마지막 노드까지 탐색해 가야 한다.
 - 따라서, 연산시간은 모두 $O(n)$ 이다.
- ② head가 리스트의 마지막 노드를 가리키는 원형연결리스트에서 삽입/삭제
 - 마지막 노드를 삭제하는 연산시간만 $O(n)$ 이고, 나머지는 $O(1)$ 이다.
 - 원형연결리스트에서 마지막 노드 삭제는 마지막 노드의 선행노드를 찾아야 하므로

기출문제 분석

1. 원형연결리스트(circular linked list)의 맨 앞에 새로운 노드를 삽입할 때, ㉠, ㉡에 대한 시간 복잡도 표현은? [2015년 국가 7급] [2020년 군무 7급]



- ① ㉠ $O(1)$ ㉡ $O(1)$
- ② ㉠ $O(1)$ ㉡ $O(n)$
- ③ ㉠ $O(n)$ ㉡ $O(1)$
- ④ ㉠ $O(n)$ ㉡ $O(n)$

2. 연결리스트에 여러 개의 노드가 존재할 때, 다음 노드를 가리키는 포인터 값이 NULL인 경우가 전혀 없는 성질을 가진 연결리스트는? [2018년 서울 7급]

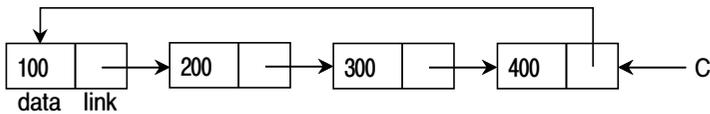
- ① 단순연결리스트(singly linked list)
- ② 이중연결리스트(doubly linked list)
- ③ 원형연결리스트(circular linked list)
- ④ 헤드노드를 가진 단순연결리스트(singly linked list with head node)

☞ 원형연결리스트

· 원형연결리스트는 마지막 노드의 포인터가 처음을 가리키므로 NULL인 포인터가 없다.

정답 : ③

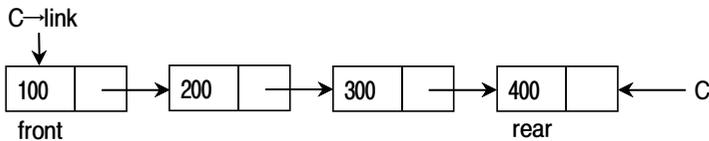
3. 다음과 같은 원형연결리스트(circular linked list) C에 대한 설명 중 가장 옳은 것은? (단, 노드는 data와 link필드로 이루어져 있고, C는 리스트의 마지막 노드를 가리키는 포인터이며, n은 노드의 개수임) [2021년 군무원 7급]



- ① 이 리스트를 큐로 사용하려면 C를 rear로 하고, C→link를 front로 한다.
- ② 이 리스트의 제일 왼쪽 노드를 제거하는데 $\Theta(n)$ 시간이 걸린다.
- ③ 이 리스트의 제일 오른쪽쪽에 새로운 노드를 삽입하는데 $\Theta(n)$ 시간이 걸린다.
- ④ C의 바로 앞 노드를 찾는데 $\Theta(1)$ 시간이 걸린다.

☞ 원형연결리스트(circular linked list) - 헤드가 리스트의 마지막을 가리키는 경우

① 이 리스트를 큐로 사용하려면 C를 rear로 하고, C→link를 front로 한다.(○)



- ② 이 리스트의 제일 왼쪽 노드를 제거하는데 $\Theta(n)$ 시간이 걸린다.(x) → $\Theta(1)$
- ③ 이 리스트의 제일 오른쪽쪽에 새로운 노드를 삽입하는데 $\Theta(n)$ 시간이 걸린다.(x) → $\Theta(1)$
- ④ C의 바로 앞 노드를 찾는데 $\Theta(1)$ 시간이 걸린다.(x) → $\Theta(n)$: 거꾸로 갈 수 없으므로

정답 : ①