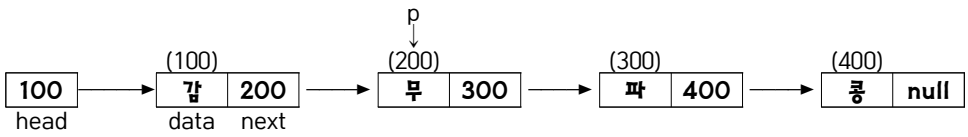
8. 단순연결리스트(singly linked list)

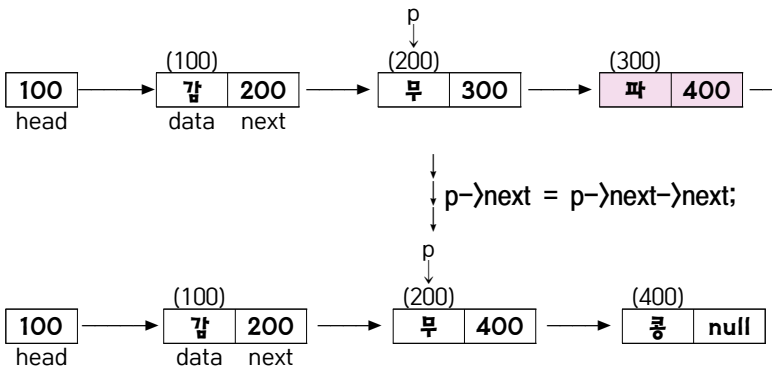
단순연결리스트의 각 노드는 하나의 포인터를 가진다.

```
struct node
{
    char data[10];
    struct node *next;   //next는 다음 노드를 가리키는 포인터
};
```



- 단순연결리스트에서는 거꾸로 가는 방법은 없다.
- 탐색은 첫 번째 노드부터 시작되어야 한다.
- 탐색시간은 포인터를 이용하여 순차적으로 추적해야 하므로 $O(n)$ 이다.

① 포인터 p가 가리키는 다음 노드 삭제



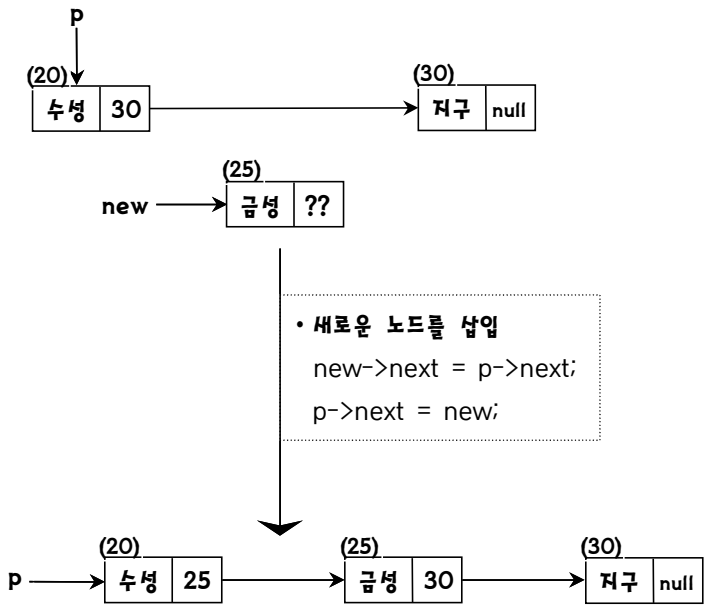
- 포인터 p가 가리키는 다음 노드 삭제 연산시간 : O(1)

[질문] 여기서, 포인터 p가 가리키는 이전 노드 삭제 연산시간은 뭘까?

- 정답은 O(n)이다. 이유는 거꾸로 갈 수 없으므로 첫 번째 노드부터 탐색되어야 한다.
- 만약, 삭제할 노드의 이전 노드 위치를 알 경우는 삭제 연산시간은 O(1)이다.

② 단순연결리스트에서 포인터 p가 가리키는 노드 다음에 새로운 노드 삽입

```
struct node
{
    char data[10];
    struct node *next;                                  //next는 다음 노드를 가리키는 포인터
};
new = (struct node *)malloc(sizeof(struct node));   //동적 메모리 할당 (힙 할당)
new->next = p->next;
p->next = new;
```



• 포인터 p가 가리키는 노드 다음에 새로운 노드 삽입 연산시간 : O(1)

기출문제 분석

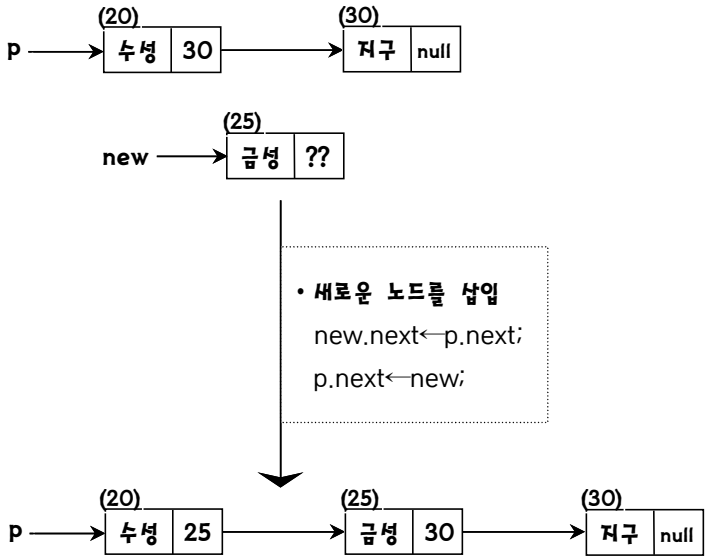
1. 단순연결리스트(singly linked list) L에서, 특정 노드 p 바로 뒤에 새로운 노드 new를 삽입하기 위한 연산 insertAfter의 의사코드가 다음과 같다. ㉠에 들어갈 내용으로 옳은 것은? [2011년 국가 7급]

```
Algorithm insertAfter(L, p, new)
  if L = NULL
    then L ← new;
    else [    ㉠    ]
  end if
  return;
```

① p.next←new; new.next←p.next;
② new.next←p.next; p.next←new;
③ new.next←p; p.next←new;
④ p.next←new; new.next←p;

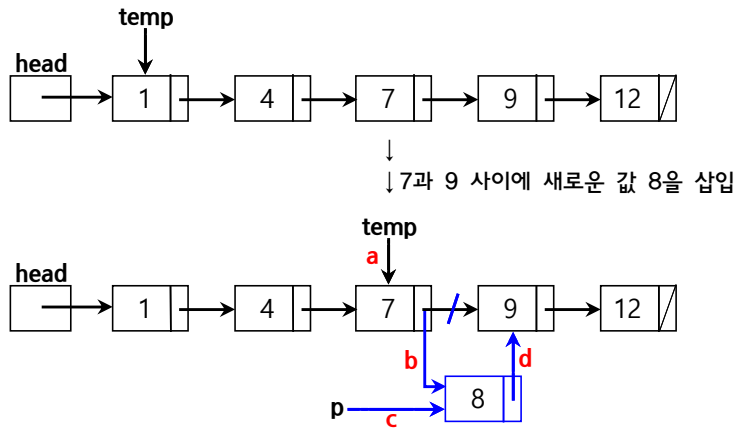
♤ 단순연결리스트에서 새로운 노드 new 삽입

• 이런 유형의 문제는 다음처럼 그림을 그려서 풀면 쉽게 답을 찾을 수 있다.



정답 : ②

2. 다음 연결리스트에서 7과 9 사이에 새로운 값 8을 삽입할 경우, 아래 a, b, c, d 연산들의 순서로 가장 옳은 것은? [2022년 군무원 7급]



---

① a → b → c → d
② a → c → b → d
③ c → a → d → b
④ c → a → b → d

♤ 연결리스트

```
// 7과 9 사이에 새로운 값 8을 삽입하는 과정을 다음과 같다.
 struct node
 {
    int data;
    struct node *next;                    //next는 다음 노드를 가리키는 포인터
 };
          ↓
          ↓노드 구조가 위와 같을 때
          ↓삽입 과정은 다음처럼 기술할 수 있다.(문제 기준)
          ↓
c : p = (struct node *)malloc(sizeof(struct node));      //삽입할 노드 메모리 할당
a : temp = 노드 7을 가리킴;      //탐색 결과 노드 7을 가리키도록 함
d : p->next = temp->next;    //삽입할 노드 8이 노드 9를 가리킴
b : temp->next = p;       //삽입할 노드 8을 가리킴
```

정답 : ③

3. n개의 데이터로 구성된 선형리스트(linear list)를 단순 연결리스트(singly linked list)로 표현하고자 한다. 다음 중 시간복잡도가 가장 낮은 연산은? [2009년 국가 7급]

① 포인터가 가리키는 노드의 다음 노드를 리스트에서 삭제
② 리스트의 길이를 출력
③ 포인터 값이 주어진 임의의 노드 앞에 새로운 노드 추가
④ 마지막 노드의 데이터를 출력

♤ **시간복잡도**

① 포인터가 가리키는 노드의 다음 노드를 리스트에서 삭제 → O(1)
② ③ ④는 O(n)이다.

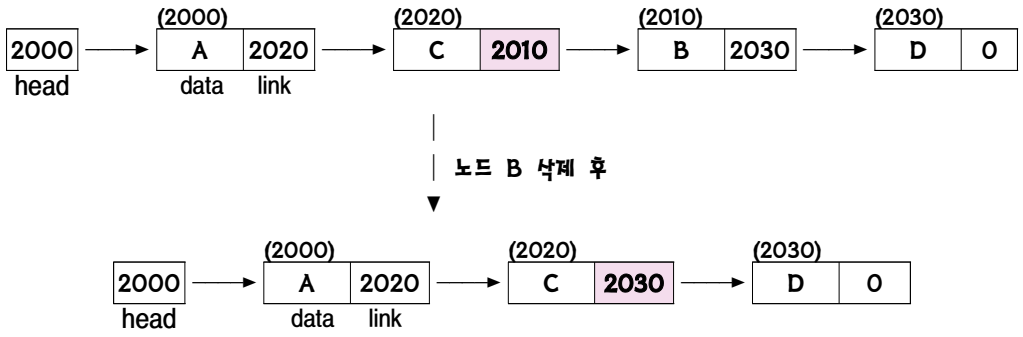
정답 : ①

4. 자료들이 단순 연결리스트(singly linked list)에 다음과 같이 구성되어 있을 때, 자료 B를 삭제한 후 변경된 내용으로 옳은 것은? [2009년 국가 7급]

메모리 주소	data	link
2000	A	2020
2010	B	2030
2020	C	2010
2030	D	0

① A의 link→2030 ② B의 link→2010
③ B의 link→2020 ④ C의 link→2030

♤ **단순 연결리스트에서 자료 삭제(그림 참조)**



정답 : ④

5. 다음의 표는 단순연결리스트(singly linked list)를 표현한 것으로 각 행은 각 노드의 주소, 데이터 및 다음 노드 주소로 구성되어 있다. <다음 노드 주소 값>은 주소가 102인 노드를 삭제한 후 다음 노드 주소의 값을 설명한 것이다. ㉠~㉣에 들어갈 값을 바르게 연결한 것은? (단, 특정 노드의 다음 노드가 없을 때 그 노드의 다음 노드 주소 값은 NULL이다) [2020년 국가 7급]

주소	데이터	다음 노드 주소
100	LEE	NULL
101	PARK	104
102	KIM	101
103	JUNG	102
104	SEO	100

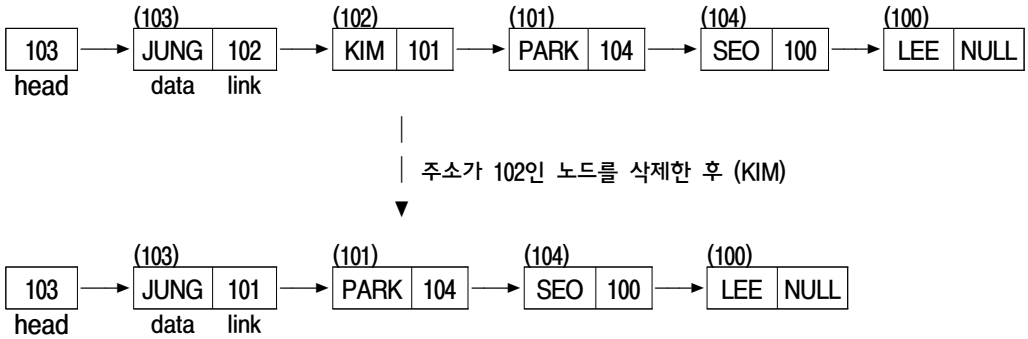
----<다음 노드 주소 값>----------

○ 주소가 100인 노드의 다음 노드 주소는 (㉠)이다.
○ 주소가 101인 노드의 다음 노드 주소는 (㉡)이다.
○ 주소가 103인 노드의 다음 노드 주소는 (㉢)이다.
○ 주소가 104인 노드의 다음 노드 주소는 (㉣)이다.

	㉠	㉡	㉢	㉣
①	NULL	104	101	100
②	NULL	103	101	104
③	101	103	102	104
④	101	104	102	100

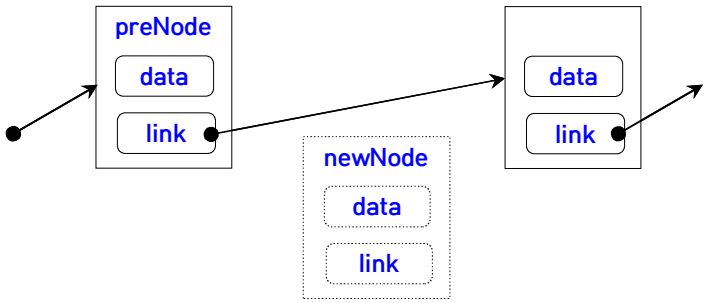
♤ 단순연결리스트에서 삭제

• 다음과 같은 단순연결리스트이다.(head의 주소는 **다음 노드 주소에 없는 값**이다)



정답 : ①

6. 연결리스트(linked list)의 'preNode' 노드와 그 다음 노드 사이에 새로운 'newNode' 노드를 삽입하기 위해 빈 칸 ㉠에 들어갈 명령문으로 옳은 것은? [2015년 국가 9급]



```
Node *newNode = (Node*)malloc(sizeof(Node));
  [          ㉠          ]
preNode->link = newNode;
```

---

① newNode->link = preNode;
② newNode->link = preNode->link;
③ newNode->link->link = preNode;
④ newNode = preNode->link;

♤ 연결리스트에서 노드 삽입

```
Node *newNode = (Node*)malloc(sizeof(Node));                    //새로운 노드 생성
newNode->link = preNode->link;          //새로운 노드 링크에 이전 노드 링크 값을 대입
preNode->link = newNode;                //이전 노드 링크에 새로운 노드 주소 값을 대입
```

정답 : ②