

### [Assignment #3]

이중 연결 리스트에서 두 노드의 위치를 교환하기

사용자로부터 자연수 리스트와 두 노드의 위치를 입력 받아, 이중 연결 리스트(Doubly Linked List)를 이용하여 두 노드의 위치를 교환하고 그 결과를 출력하는 코드를 다음과 같이 작성한다.

1. 입력 자연수의 개수는 최대 20개로 제한하고 -1 입력 시 입력을 종료한다. 자연수 간 구분은 공백으로 한다.
2. 입력한 리스트에 자연수가 아닌 것이 있으면 에러 메시지를 출력한다.
3. 입력한 자연수 리스트를 이중 연결 리스트로 구성한다.
4. 자연수 리스트 입력이 끝난 후 사용자로부터 두 자연수의 위치를 입력 받아, 이 두 숫자의 위치를 서로 바꾼다. 자연수의 위치는 0부터 시작하는 인덱스 번호로 받는다.
5. 바뀐 위치에 따라 연결 리스트를 업데이트하고, 이를 반복적으로 수행할 수 있다.
6. '종료: q' 명령이 입력될 때까지 위치 교환을 계속 수행하고, 최종 리스트를 출력한다.
7. 헤더 파일은 #include <stdio.h>, #include <stdlib.h> 두 개만 이용한다.
8. 입력한 두 숫자의 위치가 같은 경우 위치 교환을 수행하지 않고 기존 리스트를 출력한다.
9. 예외 상황인 경우 (입력한 인덱스가 범위를 초과하거나 overflow, underflow 발생 시) 위치 교환을 수행하지 않고 에러 메시지를 출력하고 기존 리스트를 출력한다.
10. 마지막으로 동적으로 할당된 모든 메모리를 해제한다.

### 실행 예

1. 자연수 리스트 입력(최대 20개, -1로 종료): 10 20 -5 40 50 60 -1  
Error: 잘못된 입력이 있습니다. 자연수만 입력해 주세요.  
자연수 리스트 입력(최대 20개, -1로 종료): 10 20 30 1.5 50 60 -1  
Error: 잘못된 입력이 있습니다. 자연수만 입력해 주세요.  
자연수 리스트 입력(최대 20개, -1로 종료): 10 20 30 40 50 60 -1  
입력 결과: 60 > 50 > 40 > 30 > 20 > 10  
위치 변경 (형식: 인덱스1 인덱스2, 종료: q): 0 4  
20 > 50 > 40 > 30 > 60 > 10  
위치 변경 (형식: 인덱스1 인덱스2, 종료: q): 5 1

20 > 10 > 40 > 30 > 60 > 50

위치 변경 (형식: 인덱스1 인덱스2, 종료: q): q

최종 결과: 20 > 10 > 40 > 30 > 60 > 50

2. 자연수 리스트 입력(최대 20개, -1로 종료): 10 20 30 40 50 60 -1

입력 결과: 60 > 50 > 40 > 30 > 20 > 10

위치 변경 (형식: 인덱스1 인덱스2, 종료: q): 3 3

60 > 50 > 40 > 30 > 20 > 10

위치 변경 (형식: 인덱스1 인덱스2, 종료: q): q

최종 결과: 60 > 50 > 40 > 30 > 20 > 10

3. 자연수 리스트 입력(최대 20개, -1로 종료): 10 20 30 40 50 60 -1

입력 결과: 60 > 50 > 40 > 30 > 20 > 10

위치 변경 (형식: 인덱스1 인덱스2, 종료: q): 1 9

Error: The specified index is out of range. Please enter a valid index.

60 > 50 > 40 > 30 > 20 > 10

위치 변경 (형식: 인덱스1 인덱스2, 종료: q): -3 12

Error: The specified index is out of range. Please enter a valid index.

60 > 50 > 40 > 30 > 20 > 10

위치 변경 (형식: 인덱스1 인덱스2, 종료: q): 2 3

60 > 50 > 30 > 40 > 20 > 10

위치 변경 (형식: 인덱스1 인덱스2, 종료: q): q

최종 결과: 60 > 50 > 30 > 40 > 20 > 10

#### 프로그램 작성 환경 조건

1. 학과 리눅스 서버 vi에디터를 이용하여 작성한 후 turnin에 제출한다.
2. gcc 컴파일러를 이용하여 컴파일한다.

#### 주의 사항

1. 미 제출시 0 점 처리, 부정행위 적발 -10점 처리함(추후 제출하는 숙제에서 감점 처리함)