

자료구조

1차 과제 - 배열구현 실행

컴퓨터공학과

2019305059

이현수

```
(1) 리스트 초기화
(2) 리스트 공백여부 확인
(3) 리스트 원소 등록
(4) 리스트 원소 개수 확인
(5) 리스트 원소 삽입
(6) 리스트 원소 삭제
(7) 리스트 원소 변경
(8) 리스트 원소 검색
(9) 프로그램 종료
--- 기능선택:1
Continue: Y
```

(1)리스트 초기화 (2)리스트 공백여부 확인

리스트를 최초 생성한 직후에 실행해야 한다.

기능 **(1)리스트초기화** 실행 시

Count = 0

순번	1	2	3	4	5	...	99	100
인덱스	0	1	2	3	4	...	98	99
데이터	x	x	x	x	x	...	x	x

```
(1) 리스트 초기화
(2) 리스트 공백여부 확인
(3) 리스트 원소 등록
(4) 리스트 원소 개수 확인
(5) 리스트 원소 삽입
(6) 리스트 원소 삭제
(7) 리스트 원소 변경
(8) 리스트 원소 검색
(9) 프로그램 종료
--- 기능선택:4
리스트에 0개 원소가 있습니다.
Continue: Y
```

리스트가 정상적으로 초기화됐는지 확인하기 위해

(4)리스트 원소 개수 확인 실행 시

리스트에 0개 원소가 있다고 정상적으로 출력된다.

Count = 0

순번	1	2	3	4	5	...	99	100
인덱스	0	1	2	3	4	...	98	99
데이터	x	x	x	x	x	...	x	x

```
(1) 리스트 초기화
(2) 리스트 공백여부 확인
(3) 리스트 원소 등록
(4) 리스트 원소 개수 확인
(5) 리스트 원소 삽입
(6) 리스트 원소 삭제
(7) 리스트 원소 변경
(8) 리스트 원소 검색
(9) 프로그램 종료
--- 기능선택:2
list empty
Continue: Y
```

기능 **(2)리스트 공백여부 확인**을 실행 시

정상적으로 "list empty"가 출력된다.

Count = 0

순번	1	2	3	4	5	...	99	100
인덱스	0	1	2	3	4	...	98	99
데이터	x	x	x	x	x	...	x	x

```
(1) 리스트 초기화
(2) 리스트 공백여부 확인
(3) 리스트 원소 등록
(4) 리스트 원소 개수 확인
(5) 리스트 원소 삽입
(6) 리스트 원소 삭제
(7) 리스트 원소 변경
(8) 리스트 원소 검색
(9) 프로그램 종료
--- 기능선택:3
현재 리스트에 0개 입력됨.
X를 입력하면 원소등록 종료
10
20
30
40
X
Continue: Y
```

(3)리스트 원소 등록 (4) 리스트 원소 개수 확인

기능 **(3)리스트 원소 등록**을 실행한다.

현재 리스트의 개수 0을 출력 후 'X'를 입력할 때까지 리스트 마지막에 등록시킨다.

10, 20, 30, 40 순서대로 리스트에 삽입된다.

Count = 4

순번	1	2	3	4	5	...	99	100
인덱스	0	1	2	3	4	...	98	99
데이터	10	20	30	40	x	...	x	x

```

(1) 리스트 초기화
(2) 리스트 공백여부 확인
(3) 리스트 원소 등록
(4) 리스트 원소 개수 확인
(5) 리스트 원소 삽입
(6) 리스트 원소 삭제
(7) 리스트 원소 변경
(8) 리스트 원소 검색
(9) 프로그램 종료
--- 기능선택:4
리스트에 4개 원소가 있습니다.
Continue: Y

```

리스트에 원소가 정상적으로 등록됐음을 확인하기 위해
기능 **(4)리스트 원소 개수 확인**을 실행하면

리스트에 4개 원소가 있다고 출력된다. 정상적으로 원소가 등록됐음을
알 수 있다.

Count = 4

순번	1	2	3	4	5	...	99	100
인덱스	0	1	2	3	4	...	98	99
데이터	10	20	30	40	x	...	x	x

```

(1) 리스트 초기화
(2) 리스트 공백여부 확인
(3) 리스트 원소 등록
(4) 리스트 원소 개수 확인
(5) 리스트 원소 삽입
(6) 리스트 원소 삭제
(7) 리스트 원소 변경
(8) 리스트 원소 검색
(9) 프로그램 종료
--- 기능선택:2
list is not empty
Continue: Y

```

리스트가 채워져 있는 상태에서
리스트 공백여부를 확인하기 위해서

(2)리스트 공백여부 확인 실행시

"list is not empty"가 정상적으로 출력된다.

Count = 4

순번	1	2	3	4	5	...	99	100
인덱스	0	1	2	3	4	...	98	99
데이터	10	20	30	40	x	...	x	x

(5) 리스트 원소 삽입

(5)리스트 원소 삽입을 실행하면 삽입할 위치, 정수값을 입력받는다.
삽입할 위치를 0으로 입력하면 "Position out of Range"가 출력되고
정수는 삽입되지 못한다.

Count = 4

순번	1	2	3	4	5	...	99	100
인덱스	0	1	2	3	4	...	98	99
데이터	10	20	30	40	x	...	x	x

```

(1) 리스트 초기화
(2) 리스트 공백여부 확인
(3) 리스트 원소 등록
(4) 리스트 원소 개수 확인
(5) 리스트 원소 삽입
(6) 리스트 원소 삭제
(7) 리스트 원소 변경
(8) 리스트 원소 검색
(9) 프로그램 종료
--- 기능선택:5
몇번째? 삽입할 정수값? 0 11
Position out of Range
Continue: Y

```

```

(1) 리스트 초기화
(2) 리스트 원소 삽입
(3) 리스트 원소 삭제
(4) 리스트 원소 변경
(5) 프로그램 종료
--- 기능선택: 5
몇번째? 삽입할 정수값? 6 11
Position out of Range
Continue: Y

```

(5)리스트원소삽입을 다시 눌러

위치값을 6으로 입력한다.

그러면 리스트에서 이격이 발생해 "Position out of Range"가 출력된다.

Count = 4

순번	1	2	3	4	5	6	...	100
인덱스	0	1	2	3	4	5	...	99
데이터	10	20	30	40	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 원소 삽입
(3) 리스트 원소 삭제
(4) 리스트 원소 변경
(5) 프로그램 종료
--- 기능선택: 5
몇번째? 삽입할 정수값? 5 50
Continue: Y

```

(5)리스트원소삽입을 다시 실행 해

위치값을 5, 삽입할 정수값을 50으로 입력한다.

Count = 5

순번	1	2	3	4	5	6	...	100
인덱스	0	1	2	3	4	5	...	99
데이터	10	20	30	40	50	x	...	x

```

(1) 리스트 초기화
(2) 리스트 원소 삽입
(3) 리스트 원소 삭제
(4) 리스트 원소 변경
(5) 프로그램 종료
--- 기능선택: 8
몇번째 원소 검색? 5
리스트 5번째 데이터 : 50
Continue: Y

```

위 단계가 정상적으로 삽입됐는지 확인하기 위해

(8)리스트원소검색을 실행해

5번째 원소를 검색한다.

결과로 50이 나오므로 정상적으로 원소가 삽입됐다.

Count = 5

순번	1	2	3	4	5	6	...	100
인덱스	0	1	2	3	4	5	...	99
데이터	10	20	30	40	50	x	...	x

(5)리스트원소삽입을 다시 실행 해

위치값을 1, 삽입할 정수값을 11으로 입력한다.

Count = 6

순번	1	2	3	4	5	6	7	...	100
인덱스	0	1	2	3	4	5	6	...	99
데이터	11	10	20	30	40	50	x	...	x

```

(1) 리스트 초기화
(2) 리스트 원소 삽입
(3) 리스트 원소 삭제
(4) 리스트 원소 변경
(5) 프로그램 종료
--- 기능선택: 5
몇번째? 삽입할 정수값? 1 11
Continue: Y

```

```

(1) 리스트 추가 기능 선택
(2) 리스트 원소 삽입 위치를 입력하세요
(3) 리스트 원소 삽입할 정수값을 입력하세요
(4) 리스트 원소 삽입할 위치를 입력하세요
(5) 리스트 원소 삽입할 정수값을 입력하세요
(6) 리스트 원소 삽입할 위치를 입력하세요
(7) 리스트 원소 삽입할 정수값을 입력하세요
(8) 리스트 원소 삽입할 위치를 입력하세요
(9) 프로그램 종료
--- 기능 선택 : 8
몇 번째 원소 검색? 1
리스트 1 번째 데이터 : 11
Continue: Y

```

위 단계가 정상적으로 동작했는지 확인하기 위해

(8) 리스트 원소 검색을 실행해

1 번째 원소를 검색한다.

결과로 11이 나오므로 정상적으로 원소가 삽입됐다.

Count = 6

순번	1	2	3	4	5	6	7	...	100
인덱스	0	1	2	3	4	5	6	...	99
데이터	11	10	20	30	40	50	x	...	x

```

(1) 리스트 추가 기능 선택
(2) 리스트 원소 삽입 위치를 입력하세요
(3) 리스트 원소 삽입할 정수값을 입력하세요
(4) 리스트 원소 삽입할 위치를 입력하세요
(5) 리스트 원소 삽입할 정수값을 입력하세요
(6) 리스트 원소 삽입할 위치를 입력하세요
(7) 리스트 원소 삽입할 정수값을 입력하세요
(8) 리스트 원소 삽입할 위치를 입력하세요
(9) 프로그램 종료
--- 기능 선택 : 5
몇 번째? 삽입할 정수값? 4 22
Continue: Y

```

(5) 리스트 원소 삽입을 다시 실행 해

위치값을 4, 삽입할 정수값을 22로 입력한다.

Count = 7

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	11	10	20	22	30	40	50	x	...	x

```

(1) 리스트 추가 기능 선택
(2) 리스트 원소 삽입 위치를 입력하세요
(3) 리스트 원소 삽입할 정수값을 입력하세요
(4) 리스트 원소 삽입할 위치를 입력하세요
(5) 리스트 원소 삽입할 정수값을 입력하세요
(6) 리스트 원소 삽입할 위치를 입력하세요
(7) 리스트 원소 삽입할 정수값을 입력하세요
(8) 리스트 원소 삽입할 위치를 입력하세요
(9) 프로그램 종료
--- 기능 선택 : 8
몇 번째 원소 검색? 4
리스트 4 번째 데이터 : 22
Continue: Y

```

위 단계가 정상적으로 동작했는지 확인하기 위해

(8) 리스트 원소 검색을 실행해

4 번째 원소를 검색한다.

결과로 22이 나오므로 정상적으로 원소가 삽입됐다.

Count = 7

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	11	10	20	22	30	40	50	x	...	x

(6) 리스트 원소 삭제

(6) 리스트 원소 삭제를 실행하면 삭제할 위치를 입력받는다.

0을 입력하면 잘못된 위치이므로 "Position out of Range"라 출력된다.

Count = 7

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	11	10	20	22	30	40	50	x	...	x

```

(1) 리스트 추가 기능 선택
(2) 리스트 원소 삽입 위치를 입력하세요
(3) 리스트 원소 삽입할 정수값을 입력하세요
(4) 리스트 원소 삽입할 위치를 입력하세요
(5) 리스트 원소 삽입할 정수값을 입력하세요
(6) 리스트 원소 삽입할 위치를 입력하세요
(7) 리스트 원소 삽입할 정수값을 입력하세요
(8) 리스트 원소 삽입할 위치를 입력하세요
(9) 프로그램 종료
--- 기능 선택 : 6
몇 번째 원소 삭제? 0
Position out of Range
Continue: Y

```

```

(1) 리스트 초기화
(2) 리스트 크기 확인
(3) 리스트 원소 개수 확인
(4) 리스트 원소 삽입
(5) 리스트 원소 삭제
(6) 리스트 원소 검색
(7) 프로그램 종료
--- 기능선택 : 6
몇번째 원소 삭제? 8
Position out of Range
Continue: Y

```

(6)리스트원소삭제를 다시 실행해

8을 입력하면 빈공간이므로 "Position out of Range"라 출력된다.

Count = 7

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	11	10	20	22	30	40	50	x	...	x

```

(1) 리스트 초기화
(2) 리스트 크기 확인
(3) 리스트 원소 개수 확인
(4) 리스트 원소 삽입
(5) 리스트 원소 삭제
(6) 리스트 원소 검색
(7) 프로그램 종료
--- 기능선택 : 6
몇번째 원소 삭제? 1
Continue: Y

```

(6)리스트원소삭제를 다시 실행해

1번째 위치를 입력하면

1번째 순번의 데이터가 삭제되고 나머지 값들이 앞으로 한칸씩 밀린다.

Count = 6

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	22	30	40	50	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 크기 확인
(3) 리스트 원소 개수 확인
(4) 리스트 원소 삽입
(5) 리스트 원소 삭제
(6) 리스트 원소 검색
(7) 프로그램 종료
--- 기능선택 : 8
몇번째 원소 검색? 1
리스트 1번째 데이터 : 10
Continue: Y

```

위 단계가 정상적으로 동작됐는지 확인하기위해

(8)리스트원소 검색을 실행해 1번째 원소를 검색하면

10이 나오므로 정상적으로 삭제됐다.

Count = 6

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	22	30	40	50	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 크기 확인
(3) 리스트 원소 개수 확인
(4) 리스트 원소 삽입
(5) 리스트 원소 삭제
(6) 리스트 원소 검색
(7) 프로그램 종료
--- 기능선택 : 6
몇번째 원소 삭제? 3
Continue: Y

```

(6)리스트원소삭제를 다시 실행해

3번째 위치를 입력하면

3번째 순번 데이터 '22'가 삭제된다.

Count = 5

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	30	40	50	x	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 크기 확인
(3) 리스트 원소 개수 확인
(4) 리스트 원소 삽입
(5) 리스트 원소 삭제
(6) 리스트 원소 변경
(7) 리스트 원소 검색
(8) 프로그램 종료
--- 기능선택: 8
몇번째 원소 검색? 3
리스트 3번째 데이터 : 30
Continue: Y

```

위 단계가 정상적으로 동작됐는지 확인하기위해

(8)리스트원소 검색을 실행해 3번째 원소를 검색하면
30이 나오므로 정상적으로 삭제됐다.

Count = 5

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	30	40	50	x	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 크기 확인
(3) 리스트 원소 개수 확인
(4) 리스트 원소 삽입
(5) 리스트 원소 삭제
(6) 리스트 원소 변경
(7) 리스트 원소 검색
(8) 프로그램 종료
--- 기능선택: 6
몇번째 원소 삭제? 5
Continue: Y

```

(6)리스트원소삭제를 다시 실행해

5번째 위치를 입력하면
5번째 순번 데이터 '50'이 삭제된다.

Count = 4

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	30	40	x	x	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 크기 확인
(3) 리스트 원소 개수 확인
(4) 리스트 원소 삽입
(5) 리스트 원소 삭제
(6) 리스트 원소 변경
(7) 리스트 원소 검색
(8) 프로그램 종료
--- 기능선택: 8
몇번째 원소 검색? 5
Position out of Range
Continue: Y

```

위 단계가 정상적으로 동작됐는지 확인하기위해

(8)리스트원소 검색을 실행해 5번째 원소를 검색하면 삭제되서 빈공간
이므로 "Position out of Range"가 출력되므로 정상적으로 삭제됐다.

Count = 4

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	30	40	x	x	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 크기 확인
(3) 리스트 원소 개수 확인
(4) 리스트 원소 삽입
(5) 리스트 원소 삭제
(6) 리스트 원소 변경
(7) 리스트 원소 검색
(8) 프로그램 종료
--- 기능선택: 7
몇번째? 변경할 정수값? 0 11
Position out of Range
Continue: Y

```

(7)리스트원소변경

(7)리스트원소변경을 실행하면 변경할 위치와 변경할 정수값을 입력받
는다. 변경할 위치를 0으로 입력하면 "Position out of Range"가 나온다.

Count = 4

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	30	40	x	x	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 원소 삽입
(3) 리스트 원소 검색
(4) 리스트 원소 변경
(5) 리스트 원소 삭제
(6) 프로그램 종료
--- 기능선택: 7
몇번째? 변경할 정수값? 5 11
Position out of Range
Continue: Y

```

(7)리스트원소변경을 다시 실행해 변경할 위치를 5로 입력하면 빈공간 이므로 "Position out of Range"가 나온다.

Count = 4

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	30	40	x	x	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 원소 삽입
(3) 리스트 원소 검색
(4) 리스트 원소 변경
(5) 리스트 원소 삭제
(6) 프로그램 종료
--- 기능선택: 7
몇번째? 변경할 정수값? 3 33
Continue: Y

```

(7)리스트원소변경을 다시 실행해 변경할 위치를 3, 변경할 정수값을 33으로 입력한다.

Count = 4

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	33	40	x	x	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 원소 삽입
(3) 리스트 원소 검색
(4) 리스트 원소 변경
(5) 리스트 원소 삭제
(6) 프로그램 종료
--- 기능선택: 8
몇번째 원소 검색? 3
리스트 3번째 데이터 : 33
Continue: Y

```

위 단계가 정상적으로 동작됐는지 확인하기위해

(8)리스트원소검색을 실행해 3번째 위치 원소를 검색하면 33이 나오므로 정상적으로 원소가 변경되었다.

Count = 4

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	33	40	x	x	x	x	...	x

(8)리스트 원소 검색

(8)리스트 원소 검색을 실행하면 몇번째 원소를 검색할지 입력받는다. 0번째 원소를 검색하기위해 0을 입력하면 "Position out of Range"가 출력된다.

Count = 4

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	33	40	x	x	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 원소 삽입
(3) 리스트 원소 검색
(4) 리스트 원소 변경
(5) 리스트 원소 삭제
(6) 프로그램 종료
--- 기능선택: 8
몇번째 원소 검색? 0
Position out of Range
Continue: Y

```



```

(1) 리스트 초기화
(2) 리스트 원소 추가
(3) 리스트 원소 삭제
(4) 리스트 원소 검색
(5) 리스트 원소 삽입
(6) 리스트 원소 변경
(7) 프로그램 종료
--- 기능선택:8
몇번째 원소 검색? 5
Position out of Range
Continue: Y

```

(8)리스트 원소 검색을 다시실행해

5번째 원소값을 검색하기 위해 5를 입력하면 빈공간이므로
"Position out of Range"가 출력된다.

Count = 4

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	33	40	x	x	x	x	...	x

```

(1) 리스트 초기화
(2) 리스트 원소 추가
(3) 리스트 원소 삭제
(4) 리스트 원소 검색
(5) 리스트 원소 삽입
(6) 리스트 원소 변경
(7) 프로그램 종료
--- 기능선택:8
몇번째 원소 검색? 4
리스트 4번째 데이터 : 40
Continue: Y

```

(8)리스트 원소 검색을 다시실행해

4번째 원소값을 검색하기 위해 4를 입력하면
40이 출력되므로 정상적으로 원소검색이 이루어진다.

Count = 4

순번	1	2	3	4	5	6	7	8	...	100
인덱스	0	1	2	3	4	5	6	7	...	99
데이터	10	20	33	40	x	x	x	x	...	x

(9)프로그램 종료

(9)프로그램 종료를 실행하면 무한반복문을 빠져나가 프로그램이 종료
된다.

```

(1) 리스트 초기화
(2) 리스트 원소 추가
(3) 리스트 원소 삭제
(4) 리스트 원소 검색
(5) 리스트 원소 삽입
(6) 리스트 원소 변경
(7) 프로그램 종료
--- 기능선택:9

```

```

C:\Users\노트북\Desktop\언어\Debug\언어.exe(프로세스
17744개)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자
동으로 닫으려면 [도구] -> [

```