

# Project Challenges - 선택정렬

**C:\W> 문제의 주제**

카드놀이 중에 카드 10장 중 아주 간단한 큰 숫자 찾기  
카드 10장을 오름차순으로 정렬하기 위한 방법에 대한 알고리즘과 순서도를 작성하기  
<45, 20, 60, 35, 10, 55, 90, 85, 75, 25>

**C:\W> 선택정렬** 을 이용하여  
알고리즘과 순서도를 표현할 것이다

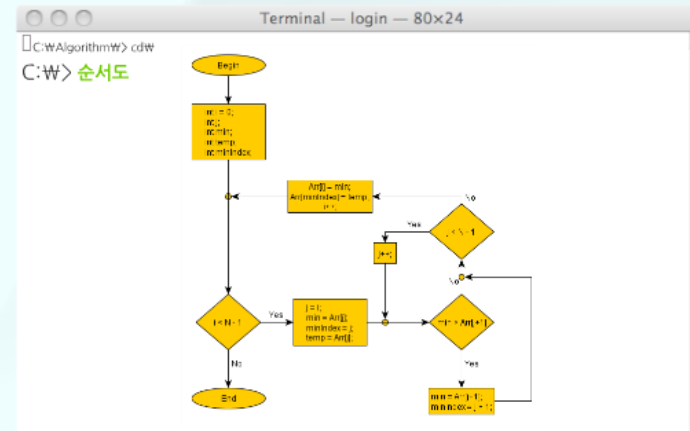
선택정렬이란? : 선택 정렬은 각 자리 정렬 알고리즘의 하나

**C:\W> 알고리즘**

· 기본 규칙

1. 루어진 리스트 중에 최소값을 찾는다.
2. 그 값을 맨 앞에 위치한 값과 교체한다(패스(pass)).
3. 맨 처음 위치를 뺀 나머지 리스트를 같은 방법으로 교체한다.

**발표 :** Algorithm 알고리즘 이규하  
flow chart 순서도 임태관



# Project Challenges - 선택정렬

**C:\W> 문제의 주제**

카드놀이 중에 카드 10장 중 아주 간단한 큰 숫자 찾기  
카드 10장을 오름차순으로 정렬하기 위한 방법에 대한 알고리즘과 순서도를 작성하기  
<45, 20, 60, 35, 10, 55, 90, 85, 75, 25>

**C:\W> 선택정렬을 이용하여 알고리즘과 순서도를 표현할 것이다**

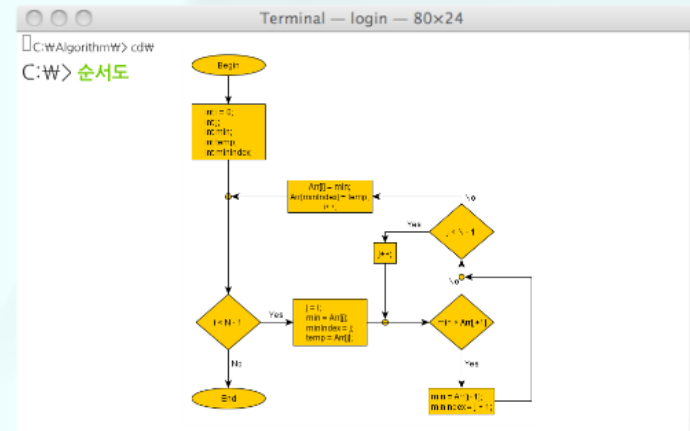
선택정렬이란? : 선택 정렬은 각 자리 정렬 알고리즘의 하나

**C:\W> 알고리즘**

· 기본 규칙

1. 루어진 리스트 중에 최소값을 찾는다.
2. 그 값을 맨 앞에 위치한 값과 교체한다(패스(pass)).
3. 맨 처음 위치를 뺀 나머지 리스트를 같은 방법으로 교체한다.

**발표 :** Algorithm 알고리즘 이규하  
flow chart 순서도 임태관



□ C:\W> 문제의 주제

카드놀이 중에서 카드 10장 중  
아주 간단한 큰 숫자 찾기

카드 10장을 오름차순으로 정렬하기 위한  
방법에 대한 알고리즘과 순서도를 작성하기

<45, 20, 60, 35, 10, 55, 90, 85, 75, 25>

```
C:\> cd Program Files (x86)\Microsoft Visual Studio 12.0\
```

```
C:\Program Files (x86)\Microsoft Visual Studio12.0\> 우리팀이 선택한 정렬 알고리즘
```

**선택정렬** 을 이용하여

알고리즘과 순서도를 표현할 것이다

선택정렬이란? : 선택 정렬은 제자리 정렬 알고리즘의 하나

```
C:\W> cd Algorithm
C:\WAlgorithm> tree
```

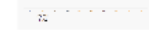
C:\W> 알고리즘

## • 기본 규칙

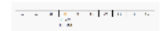
1. 주어진 리스트 중에 최소값을 찾는다.
2. 그 값을 맨 앞에 위치한 값과 교체한다(패스(pass)).
3. 맨 처음 위치를 뺀 나머지 리스트를 같은 방법으로 교체한다.



• 맨 처음에 있는 수는 정렬이 완료되었으니 다음 수부터의 정렬  
 • 전체 리스트의 수를 확인하고 (20) 수를 정렬하기 위한 작업을 수행한다 (20 < 30)  
 • 나머지 리스트에 대해 반복해서 같은 작업을 수행한다



• 맨 처음에 있는 수는 정렬이 완료되었으니 다음 수부터의 정렬  
 • 전체 리스트의 수를 확인하고 (20) 수를 정렬하기 위한 작업을 수행한다 (20 < 30)  
 • 나머지 리스트에 대해 반복해서 같은 작업을 수행한다



• 맨 처음에 있는 수는 정렬이 완료되었으니 다음 수부터의 정렬  
 • 전체 리스트의 수를 확인하고 (20) 수를 정렬하기 위한 작업을 수행한다 (20 < 30)  
 • 나머지 리스트에 대해 반복해서 같은 작업을 수행한다



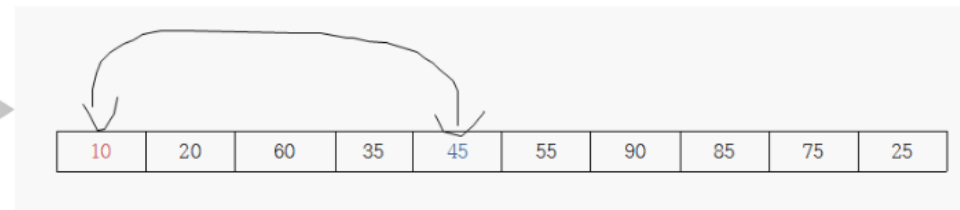
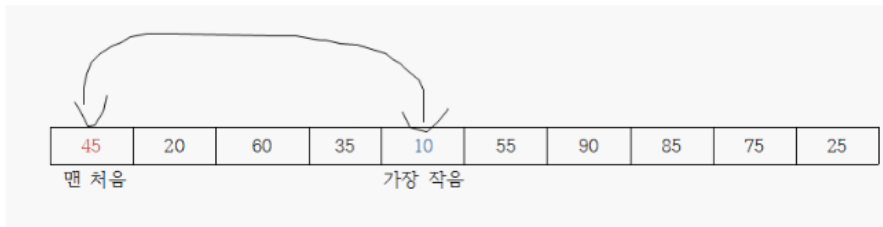
• 맨 처음에 있는 수는 정렬이 완료되었으니 다음 수부터의 정렬  
 • 전체 리스트의 수를 확인하고 (20) 수를 정렬하기 위한 작업을 수행한다 (20 < 30)  
 • 나머지 리스트에 대해 반복해서 같은 작업을 수행한다



## • 기본 규칙

1. 주어진 리스트 중에 최소값을 찾는다.
2. 그 값을 맨 앞에 위치한 값과 교체한다(패스(pass)).
3. 맨 처음 위치를 뺀 나머지 리스트를 같은 방법으로 교체한다.

- 처음부터 끝에 있는 수를 모두 스캔해서 가장 작은수를 선택함 (수 : 10)
- 선택한 수(10)와 맨 처음에 있는 수(45)를 바꿈



- 맨 처음에 있는 수는 정렬이 완료되었으니 남은 수들끼리 정렬
- 전에 이동한 수를 제외하고 남은 수들을 스캔해서 가장 작은수를 선택함 (수 : 20)
- 선택한 수(20)와 맨 처음의 수(20)를 바꿈, 하지만 20이 처음수이므로 그대로 정렬

10	20	60	35	45	55	90	85	75	25
----	----	----	----	----	----	----	----	----	----

가장 작음  
맨 처음





- 맨 처음에 있는 수는 정렬이 완료되었으니 남은 수들끼리 정렬
- 전에 이동한 수를 제외하고 남은 수들을 스캔해서 가장 작은수를 선택함 (수 : 35)
- 선택한 수(35)와 맨 처음의 수(35)를 바꿈, 하지만 35이 처음수이므로 그대로 정렬

10	20	25	35	45	55	90	85	75	60
----	----	----	----	----	----	----	----	----	----

맨 처음  
가장 작음

- 맨 처음에 있는 수는 정렬이 완료되었으니 남은 수들끼리 정렬
- 전에 이동한 수를 제외하고 남은 수들을 스캔해서 가장 작은수를 선택함 (수 : 45)
- 선택한 수(45)와 맨 처음의 수(45)를 바꿈, 하지만 45이 처음수이므로 그대로 정렬

10	20	25	35	45	55	90	85	75	60
----	----	----	----	----	----	----	----	----	----

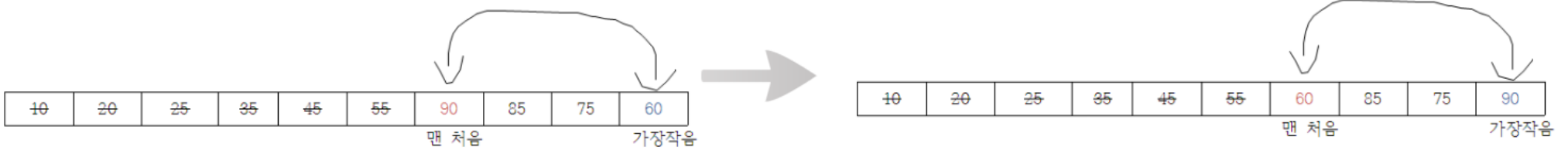
맨 처음  
가장 작음

- 맨 처음에 있는 수는 정렬이 완료되었으니 남은 수들끼리 정렬
- 전에 이동한 수를 제외하고 남은 수들을 스캔해서 가장 작은수를 선택함 (수 : 55)
- 선택한 수(55)와 맨 처음의 수(55)를 바꿈, 하지만 55이 처음수이므로 그대로 정렬

10	20	25	35	45	55	90	85	75	60
----	----	----	----	----	----	----	----	----	----

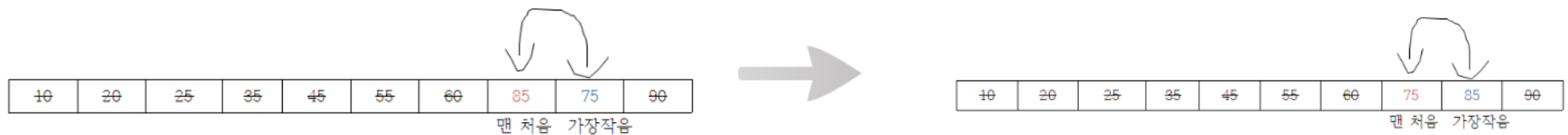
맨 처음  
가장 작음

- 맨 처음에 있는 수는 정렬이 완료되었으니 남은 수들끼리 정렬
- 전에 이동한 수를 제외하고 남은 수들을 스캔해서 가장 작은수를 선택함 (수 : 60)
- 선택한 수(60)와 맨 처음에 있는 수(90)를 바꿈



?

- 맨 처음에 있는 수는 정렬이 완료되었으니 남은 수들끼리 정렬
- 전에 이동한 수를 제외하고 남은 수들을 스캔해서 가장 작은수를 선택함 (수 : 25)
- 선택한 수(25)와 맨 처음에 있는 수(60)를 바꿈



C:\WAlgorithmW> cdW

C:\W> 순서도

