

## 과제 3. 정렬

# 과제 3

- ◆ 제출 마감일 : 4월 27일 (화요일) 10:59pm
- ◆ eClass 과제방에 제출
- ◆ 사용 언어
  - C / Java / Python / C++ 중 본인이 선택
- ◆ 제출 양식
  - 보고서 (표지 + 문제 내용과 해결 방안 + 결과 캡처)
  - 소스코드 (별도 파일로 제출)
- ◆ 보고서는 Word/HWP/PPT/PDF 로 제출
- ◆ 1인 과제

타인의 과제를 복사하지 말 것

# (1) 대량의 데이터 생성하기

## ◆ 다음의 구성을 가진 데이터 5만 명분을 랜덤하게 생성한다.

- 학번(13-21년도에 입학한 학생의 8자리 숫자로 랜덤하게 생성), 이름(10자의 영문자로 랜덤하게 생성), 전화번호(010 으로 시작하는 11자리 숫자로 랜덤하게 생성)
- 동일한 학번을 가진 학생은 없어야 한다. 이름은 같을 수 있다. 전화번호도 같을 수 있다.
- 동일한 학번을 가진 학생이 없다는 것을 어떻게 보증할 것인지 설명하라. 이 때의 시간 복잡도는 어떻게 되나? (보고서에 설명)

## (2) 내장된 정렬 방법과 비교하기

- ◆ 각 언어(C, Java, Python)에는 내장된 정렬 방법이 있다.
  - (1)에서 만든 데이터를 해당 언어가 가진 정렬 함수로 정렬해 보자.
  - (모든 정렬은 오름 차순) : 작은 값에서 큰 값 순으로 정렬
  - 정렬의 기준은 이름
  - 이 때 걸리는 시간을 측정해 보자.

### (3) 선택 정렬/퀵 정렬/힙 정렬

- ◆ 동일한 데이터에 대해 선택 정렬, 퀵 정렬, 힙 정렬을 수행한 후 결과를 비교하자.
  - 데이터가 달라지면 결과에 영향을 줄 수 있으니 (2), (3)은 모두 동일한 데이터로 실행해야 한다.
  - 정렬의 기준은 이름으로 한다. (모든 정렬은 오름 차순)
  - 5만 명의 데이터를 모두 화면에 출력하면 시간이 너무 오래 걸리니 결과 확인은 1000명당 1개만 화면에 표시한다.
    - 1번, 1001번, 2001번, 3001번, ..., 49001번 (총 50명)
  - 정렬이 잘 되었다는 것을 어떻게 보증할 것인지 설명하라. 이것의 시간 복잡도는 어떻게 될까? (보고서에 설명)
  - 각 정렬에 소요되는 시간이 얼마나 되는지 (2), (3)의 각 방법을 비교한다.