

1. Problem Statement

LMS 'Assignment 01'에 'birthday.in' 파일을 이용하여 학생들의 생년월일 데이터를 정렬한다. 정렬은 '월' 데이터를 기준으로 병합 정렬, 또는 기본 퀵 정렬 알고리즘을 사용해 오름차순으로 정렬한다. 입력 파일의 크기가 작고 단순하기 때문에 입력 파일은 학생이 직접 문제에 맞춰 파악하여 알고리즘을 작성한다.

[입력]

입력 파일의 이름은 'birthday.in'이다. 학생 한 명의 이름과 생일 정보가 각각 한 줄에 공백으로 구별되어 있다. 데이터의 개수는 주어지지 않는다.

[출력]

출력 파일의 이름은 'birthday.out'이다. 정렬한 데이터는 학생 한명을 각각 한 줄에, 그리고 이름과 생일은 공백으로 구별하여 출력한다.

[입출력 예]

입력 (그림 1의 그래프)

```
Yeji 1201
Suji 0804
Sujeong 0905
```

출력

```
Suji 0804
Sujeong 0905
Yeji 1201
```

2. Restriction

이번 과제에서 제출하는 소스 코드에는 병합 정렬과 기본 퀵소트 알고리즘 중 하나만 사용한다. 이번 과제를 포함, 알고리즘 강좌의 나머지 숙제에서 제출할 프로그램은 반드시 CPU 상에서

single thread로 수행되어야 한다. 멀티코어 혹은 멀티쓰레드 컴퓨팅은 허용하지 않으며, gpu 활용 역시 허용하지 않는다. 프로그램은 외부와 통신 없이 단독으로 수행되어야 한다. 앞의 제약을 위반한 경우 큰 패널티를 받는다.

3. Language

프로그램 언어는 C++, JAVA, 파이썬 중 하나를 사용한다. 기본적으로 제공하는 라이브러리 이외의 다른 라이브러리를 사용하는 경우에는 미리 교수와 상의해야 한다. 교수와 상의 없이 다른 언어나 라이브러리를 사용하는 경우, 큰 페널티를 받을 수 있다.

4. Compilation and Execution

제출 시 반드시 Makefile을 포함해야 하며, make all 명령으로 컴파일이 가능하도록 만들어야 한다. C++을 사용하는 경우 컴파일러의 최적화 옵션은 -O3까지 사용할 수 있다. 또한 make run 명령으로 프로그램이 수행되도록 한다. 아래 예시와 같이 명령을 입력했을 때, 여러분이 구한 해가 birthday.out에 기록되어야 한다.

```
$ ls
Makefile yourfile1.cpp yourfile2.cpp ...

$ make all

...

$ cp /home/alg_Ta/instance1 ./birthday.in

$ make run

...

$ cat ./birthday.out

Suji 0804
Sujeong 0905
Yeji 1201
```

컴파일과 수행은 리눅스에서 가능해야 한다. 채점 컴퓨터는 Intel Xeon Scalable 4210 이며, OS는 Ubuntu Server 22.04.4 LTS다.

4. Submission

4.1. 압축 파일명 및 제출물

2022-1234.* (압축 확장자)

- zip, tar, tar.gz
- 중간에 하이픈 필수(-)

압축 시 포함 항목

- Makefile
- 소스코드
- PDF 형식의 보고서 (2022-1234.pdf)

4.2. 압축 해제 시 폴더명

압축을 풀었을 때 압축 해제한 폴더의 이름이 자신의 학번명이 되도록 한다. 예를 들어, 압축 해제 시 2022-1234 폴더가 아닌 hw1 등으로 압축이 풀리면 다른 수강생 과제와 충돌이 생기는 경우가 발생하게 된다. 압축 하기 전에 hw1 폴더를 2022-1234 로 바꾼 후 압축한다.

4.3. Makefile / birthday.in / birthday.out 의 위치

- 2022-1234/Makefile
- 2022-1234/birthday.in
- 2022-1234/birthday.out

Makefile / birthday.in / birthday.out 의 위치를 항상 학번 폴더의 최상단에 위치하게 한다. 제출할 때 birthday.in을 포함할 필요는 없지만, 제출한 프로그램이 해당 위치에 있는 birthday.in을 입력으로 받을 수 있어야 한다.

채점 시 수강생들의 압축파일들을 모아놓고 한번에 풀게 되는데, 이 때 수강생 각각의 학번으로 된 폴더가 생성되게 된다. 각각의 학번 폴더로 진입하자 마자, 채점용 인스턴스 파일이 복사된다("2022-1234/birthday.in"). "make all"과 "make run"을 수행한 후 그 결과로 현재 위치에 생성된 "2022-1234/birthday.out"을 수집한다.

오류 예시:

오류 1. 2022-1234/src/Makefile
오류 2. 2022-1234/data/birthday.in
오류 3. 2022-1234/output/birthday.out

4. Team

2024년 알고리즘 수업 과제는 모두 개인 과제다.

5. Report

프로젝트에 대해 보고서를 작성하여 제출한다. 분량은 A4 2페이지 이하다. 보고서는 논문 형식으로 작성한다. 보고서 내에 소스 코드를 그대로 포함해서는 안된다. 포함 시 큰 패널티를 받는다. 보고서 본문에는 최소한 다음 내용을 포함시킨다.

- 자료구조 수업 자료를 참고하여 병합 정렬과 기본 퀵 정렬 알고리즘 소개
- 두 정렬 방법을 이용해 birthday.in을 정렬한 결과 비교
- 제출한 소스코드에 사용한 정렬 알고리즘과 선택 이유
- 컴퓨터 프로그래밍 없이 "같은 생일자를 찾는 알고리즘"을 정렬하기 전과 정렬한 후로 나누어 설명
- 아래 알고리즘들의 correctness와 efficiency를 보일 것
 1. (정렬 전 데이터에 대해) 같은 생일자를 찾는 알고리즘
 2. (정렬 후 데이터에 대해) 같은 생일자를 찾는 알고리즘
 3. 병합 정렬
 4. 기본 퀵 정렬

6. Grading

개개인의 프로그램 수행 시간은 최대 1초다. 제출한 모든 파일은 점수와 함께 LMS에 공개된다. 지각 제출은 불허하고, 제출 형식을 지키지 않을 경우 0점 처리한다.