

# HW 2

(프로듀스 N)

Kyu-haeng Lee

Fall, 2018

# 개요

## ■ 배경

- 대국민 아이돌 선발 프로젝트 "프로듀스 N"의 김작가는 어느날 출연자 정보를 기록해 놓은 파일이 심각하게 훼손되었음을 발견한다. 파일을 복구해서 김작가를 도와주자.

## ■ 문제설명

- 훼손된 파일이 주어진다. [파일이름]은 정해져 있지 않다.
- 훼손된 파일로부터 유효한 정보를 추출하고 정리하여 [파일이름].recovery로 저장한다.

## ■ 제출물 (due date: 10/19 9PM)

- recovery.c (혹은 recovery.cpp)

# 파일정보

## ■ 원본 파일정보

- 한줄마다 출연자의 정보가 들어있다.
  - ✓ 20줄이 있다면 총 20명의 출연자가 있다는 뜻이다.
- 각 출연자 정보는 5개의 **속성**(이름, 나이, 키, 점수, 등급)으로 구성된다.
  - ✓ 각각의 속성은 **큰 따옴표로 둘러싸여있고 대문자 속성이름과 콜론으로 시작하는 일정한 형식**을 갖추고 있다. 즉, "[속성이름]: [값]"의 형태다. 아래 그 예를 보여준다.
    - "NAME: G-Dragon"
    - "AGE: 19"
    - "HEIGHT: 180"
    - "POINT: 86"
    - "CLASS: A"
- CLASS는 A, B, C, F 4등급이 있다.
- 각 속성의 길이는 30bytes이하다.
- 노이즈가 끼면서 문자열이 늘어날 수 있다. 이때 한 줄의 길이는 300bytes 이하다.

# 예시

## ■ 원본 파일 예시

```
"NAME: G-Dragon" "AGE: 19" "HEIGHT: 180" "POINT: 86" "CLASS: C"
```

```
"NAME: IU" "AGE: 29" "HEIGHT: 155" "POINT: -30" "CLASS: A"
```

- 참가자의 수가 2명이며, 각각의 속성을 볼 수 있다.

## ■ 훼손된 파일 예시

```
`12"CLASS2: abc"kl5`ew"AGE: -19"[=-"HEIGHT: 0180""12"NAME: G-Dragon"h"POINT: 86"12bgrweh"CLASS: Ccaqw"235`bq
```

```
"CLASS: a"qtqt"NAME: IU"qqt`21"AGE: -029" 1bwe2415"HEIGHT: 000155"nwnw "POINT: -30"
```

- 원본파일이 훼손되면서 다음 현상이 발생했다.
  - ✓ 노이즈가 끼면서 문자열이 늘어난다.
  - ✓ 속성 이름 및 값에 변형이 생겼다.
  - ✓ 속성의 위치가 변경되었다.

# 노이즈

## ■ 노이즈

- "" 외부 노이즈

- ✓ 이 경우는 무조건 노이즈이므로 제거 대상이다.

- "" 내부 노이즈

- ✓ ""내부에 생기는 노이즈는 ""안의 속성이름과 값을 변형시킬 수 있다. 속성이름이 변형된 경우는 해당 속성을 버린다. 속성 값만 변형된 경우는 복구해서 쓴다.

- ✓ 1) 속성이름 변경

- 대문자여야하는 속성이름의 일부를 소문자로 바꾼다. ("NAME: " → "nAme: ")
    - 숫자가 속성이름에 추가된다. ("NAME: " → "nAme2: ")
    - 정의되어 있지 않은 속성이름이 추가된다. ("WEIGHT: 45")
    - 이 경우는 노이즈이므로 제거해야한다.

- ✓ 2) 속성 값 변경

- 숫자앞에 -/+ 혹은 0이 추가된다. ("AGE: 10" → "AGE: -010")
    - CLASS 값을 소문자로 바꾸거나 뒤에 쓰레기값이 붙는다. ("CLASS: B" → "CLASS: bCFFF")
    - 속성이름은 올바르나 값만 올바르지 않은 경우는 값만 복구하면 유효한 속성이 된다.

## ■ 한 줄에 유효한 속성정보 (즉, ""로 둘러싸인 것)가 5개가 있음이 항상 보장된다.

# 요구사항 (일반)

## ■ 프로그램 실행방법

- 가정
  - ✓ 주어진 파일을 info라고 하자
  - ✓ recovery.c를 빌드한 프로그램을 myprog라고 하자.
- 즉, 현재 폴더안에 적어도 다음 3개의 파일이 존재한다.
  - ✓ info
  - ✓ recovery.c
  - ✓ myprog
- 실행할때 파일이름을 인자로 받는다.
  - ✓ \$) ./myprog info
- 실행하면, 1) 먼저 info의 내용이 그대로 stdout에 출력되고, 2) 복구된 파일인 info.recovery가 생성된다.
  - ✓ 이제 폴더 안에 적어도 다음 4개의 파일이 존재한다.
    - info
    - recovery.c
    - myprog
    - info.recovery

# 요구사항 (일반)

## ■ Error 처리 함수 사용

- perror 혹은 strerror 함수를 한 개 이상 사용해서 에러가 발생했을때 함수가 호출되도록 해라.
- 참고로, 프로그램 실행 도중 에러가 없다면 위 함수가 호출되지 않는 게 당연하다.

# 요구사항 (기능 1)

## ■ 파일내용 출력

- 기능 1은 파일의 내용을 “있는 그대로” stdout으로 출력한다.

## ■ 예시

- 파일이름이 abc.txt라고 하자.
  - ✓ 참고: 이때 복구된 파일이름은 abc.txt.recovery가 된다.
- ```
$)./myprog abc.txt
QR$S"NAME: G-DRAGON"@#%2R$S"AGE: 23"@#%WETa1251q
"CLASS: A"235R$S"AGE: 31"@#%WE"NAME: MR.P"T]p

$)
```



# 요구사항 (기능 2)

## ■ 노이즈 제거

- 외부 노이즈 제거
  - ✓ 노이즈 제거 전) !@\$QR\$S"NAME: G-DRAGON"@#%WETasqwrjrt3\$
  - ✓ 노이즈 제거 후) "NAME: G-DRAGON"
- 내부 노이즈 제거
  - ✓ 내부 노이즈는 속성 이름이 올바르지 않은 것을 의미한다.
  - ✓ "name: G-DRAGON" (name이 소문자)
  - ✓ "AGE2: 33" (AGE옆에 숫자)
- 이 종류의 노이즈는 프로그램 내에서 무시하면 된다.

# 요구사항 (기능 3)

## ■ 속성 값 보정

- 속성 이름을 유효하지만, 값이 오류가 있을 경우 다음 룰에 따라 보정한 다.
  - ✓ AGE: 아래 세 종류의 오류가 발생할 수 있다.
    - "AGE: 0019" → 19 (앞에 0이 있는 경우 0을 무시한다.)
    - "AGE: -19" → 19 (앞에 -부호가 있는 경우 -를 무시한다.)
    - "AGE: +19" → 19 (앞에 +부호가 있는 경우 +를 무시한다.)
  - ✓ HEIGHT: AGE와 동일하게 세 종류의 오류가 발생할 수 있고, 보정룰도 동일하다.
  - ✓ CLASS: 아래 두 종류의 오류가 발생할 수 있다.
    - "CLASS: a" → A (소문자인 경우 대문자로 바꾼다.)
    - "CLASS: bAAFFB" → B (첫번째 문자만 남기고 버린다.)

# 요구사항 (기능 4)

## ■ CLASS 업데이트

- A, B, C, F 4개의 CLASS가 있고, 각 출연자의 CLASS는 POINT값에 따라 업데이트 되어야한다. 각 CLASS의 인원은 모두 같다.
  - ✓ 총 20명이라면, 각 CLASS에 5명씩 할당된다.
  - ✓ 출연자의 수가 항상 4의 배수가 됨을 보장한다.
- POINT값을 기준으로 CLASS를 업데이트해라.

# 요구사항 (기능 5)

## ■ 복구 파일 생성

- [파일이름].recovery에는 크게 2가지 정보가 포함된다.
  - ✓ info.recovery 참고
- 1) 출연자 정보
  - ✓ 총 출연자 수를 표시한다.
  - ✓ 출연자 정보를 쓸 때 앞에 인덱스를 붙여준다.
  - ✓ 속성은 NAME, AGE, HEIGHT, POINT, CLASS 순으로 쓴다.
  - ✓ POINT가 높은 출연자부터 내림차순으로 쓴다.
  - ✓ CLASS의 경우 이동 결과를 표시한다.
    - CLASS: A (<-- B)
- 2) 통계
  - ✓ 출연자 정보 밑에 나이, 키에 대한 출연자 통계를 쓴다.
  - ✓ 소수 두번째자리까지 표시한다.
  - ✓ avg. age: 33.99
  - ✓ avg. height: 178.09

# 채점기준

|      |                                        |     |
|------|----------------------------------------|-----|
| 기본   | strerror 혹은 perror이 하나 이상 적절히 포함되어 있다. | 1   |
|      | 빌드/실행 에러가 없다.                          | 1   |
|      | 실행하면 복구 파일이 생성된다.                      | 0.5 |
|      | 복구 파일 이름이 적합하다.                        | 0.5 |
|      | 합계                                     | 3   |
| 기능 1 | 실행하면 파일 내용이 stdout에 출력된다.              | 2   |
|      | 합계                                     | 2   |
| 기능 2 | 외부 노이즈가 모두 제거된다.                       | 2   |
|      | 내부 노이즈가 모두 제거된다.                       | 2   |
|      | 합계                                     | 4   |
| 기능 3 | 속성 값이 모두 보정된다.                         | 2   |
|      | 합계                                     | 2   |
| 기능 4 | CLASS가 올바르게 업데이트된다.                    | 2   |
|      | 합계                                     | 2   |
| 기능 5 | 참여자 정보가 모두 표시된다.                       | 0.5 |
|      | 참여자의 각 속성이 정렬되어 있다.                    | 0.5 |
|      | 총 참여자 수가 표시된다.                         | 0.5 |
|      | 참여자 인덱스 정보가 표시된다.                      | 0.5 |
|      | 참여자 정보가 POINT 순으로 내림차순 정렬되어 있다.        | 2   |
|      | 참여자의 CLASS 이동 정보가 나타난다.                | 1   |
|      | 나이 통계가 소수점 2자리 까지 정확히 표시된다.            | 1   |
|      | 키 통계가 소수점 2자리까지 정확히 표시된다.              | 1   |
|      | 합계                                     | 7   |
|      | 합계                                     | 20  |