과제 **6**

문제풀이 보고서

2023100085 신희원
2023100086 전하은
2023100096 김익희

1. 파트 분배 과정

팀원이 세명이니 과제설명 pdf에 나온 그대로 데이터 추출 파트(A), 그래프 구현 파트 (B), 메인 모듈 작성 파트(C) 이렇게 세 파트로 설계함. 각각 김익희 (2023100096), 신희원 (2023100085), 전하은(2023100086) 이 전담하기로 결정하였음.

2. A_part의 반환 값 설정 (build_data.py)

1. 사전에 나는 대화를 기반으로 아래 사진과 같은 형식으로 데이터를 반환하기로 함.

```
"과목명1": {
    "표준점수1": [남학생수1, 여학생수1],
    "표준점수2": [남학생수2, 여학생수2],
    ...
},
```

2. **csv** 모듈 **import**
   - csv 파일 속의 데이터를 효율적으로 다루기 위해 csv 모듈 불러오기
3. 변수 설정 및 파일 불러오기
   - 추출한 데이터를 저장할 딕셔너리인 “data”를 생성
   - open 메서드를 이용하여 csv 파일을 불러오기
   - csv 모듈에 있는 reader를 사용하여 csv 파일의 각 행을 리스트로 변환하여 각의 줄별로 reader에 데이터가 저장됨
4. 사전에 계획한 형식대로 데이터 추출
   - 반복문을 사용하여 모든 행과 열에 있는 데이터를 갖고 온다
   - data[line[1]] : 이중 딕셔너리로 반환할 것이기 때문에 key값이 과목명인 딕셔너리를 생성
   - 안쪽 딕셔너리는 점수가 key값이 되도록 하고, value값은 남학생 수(line[3]), 여학생 수(line[4])가 리스트의 형식으로 들어가도록 딕셔너리 업데이트
   - 각 열에 저장된 데이터를 점수와 인원은 정수형이 되도록 딕셔너리에 저장
5. 사전에 약속된 형식으로 추출한 **data** 딕셔너리를 반환하고 함수**(build_data)** 종료

3. B_part의 매개변수 설정 (Draw_graph.py)

일단 그래프를 그리기 위해 받아야 하는 데이터는 1)과목 이름 2) 표준점수에 따른 남학생 분포 3) 표준점수에 따른 여학생 분포 이렇게 세가지 정보가 필요하다. 이중 2) 3)의 데이터를 어떤 형식을 받아야 하는지 결정해야 했다. 결과적으로 딕셔너리를 이용해 받기로 설정하였다. 그 이유는 아래와 같다.

파이썬에서 matplotlib를 활용해 우리가 원하는 순차적으로 이어지는 프로그램을 만들기 위해서는 해당 조건들을 차례로 만족해야 한다. (여기서 x는 표준점수를 y는 학생 수로 설정 하였다.)

1) x배열의 인덱스 수와, y배열의 인덱스 수가 같아야한다.
2) x의 데이터가 순차적이어야 한다. (그렇지 않으면 지그재그로 요동치는 그래프를 그리게 됨)
3) x배열 속 값이 y배열과 대응되는 값이어야 한다.

다음을 만족하기 위해 {표준점수:학생 수}로 된 딕셔너리를 파라미터로 받아 표준점수(key)값을 담은 배열을 먼저 추출 한 후, 해당 배열을 오름차순 순서대로 정리한 것을 x배열에 대입해 준다. 그리고 그 x배열의 순서대로 딕셔너리에서 학생 수를 추출해 y배열에 대입해준다. 이렇게 하면 위 세가지 조건을 모두 만족하는 형태의 데이터가 만들어 진다. 이렇게 남학생과 여학생 총 4개의 데이터 배열을 만들어 주고 해당 데이터를 matplotlib에 있는 plot매소드의 파타미터로 전달 해 준다.

4. C_part의 메인모듈 설계 (main.py)

1. **build_data.py**와 **Draw_graph.py** 파일을 **import**해서 불러오기
2. 사용자에게 과목명 입력받기
   - 사용자로부터 입력받은 과목명에 해당하는 변수 : user_choice
   - input 메서드 이용
3. 데이터 불러오기
   - build_data.py에 구현되어 있는 데이터 추출 함수(build_data) 사용
   - 사용자가 입력한 과목명과 대조를 위해 추출한 데이터에서 과목명 추출 (과목명은 겉 딕셔너리의 key값 → .keys() 이용)
4. 과목 찾기
   - 사용자가 입력한 과목을 추출한 데이터에서 찾기 위해, 딕셔너리에서 추출한 key값의 리스트를 반복문을 사용하여 하나씩 탐색
   - 만약, 사용자가 입력한 과목명을 찾았다면, 해당 과목명을 draw_subject 라는 변수에 저장하고 반복문 종료
   - 예외처리 : 만약, 입력한 과목명이 데이터에 없다면, 그 사실을 출력하고 프로그램 종료
5. **draw_grapg.py** 파일에 있는 **DrawG** 함수의 매개변수 형식을 맞추기 위해 데이터 추출
   - 점수 추출 : 안쪽 딕셔너리의 key값
   - 남자 점수-인원 딕셔너리 추출 : 점수 리스트 중에서 첫 번째 요소가 남자 인원이 되므로 “male”이라는 딕셔너리를 생성한 후, key값은 점수(score), value값은 인원(안쪽 딕셔너리의 value값인 리스트의 첫 번째 요소)이 되도록 반복문을 사용하여 딕셔너리에 하나씩 추가
   - 여자 점수-인원 딕셔너리 추출 : male 딕셔너리와 동일한 방법으로 female 딕셔너리 형성
6. 그래프 그리기
   - Draw_graph.py 파일 속에 있는 DrawG 함수 이용

5. 프로그램 실행 후 오류 (혹은 결함) 발견 및 수정

- 인코딩 문제 : 파일의 인코딩 형식이 무엇인지 파악하는 것이었습니다. 인코딩은 리눅스의 file 명령어를 통해, 아래와 사진과 같이 확인
  그러나, 인코딩을 iso-8859-1로 해서 코드를 실행해도 한글이 깨져서 추출되는

```
ikhee@DESKTOP-P17FP62:/mnt/c/Users/kimik/python$ file -i 20231231.csv
20231231.csv: text/csv; charset=iso-8859-1
```

  문제가 발생함. 이에, gpt에게 분석을 부탁했더니 파일이 **iso-8859-1** 이 아니라 **euc-kr** 로 작성되었다는 것을 확인할 수 있었음. 인코딩을 해당 형식으로 바꾸었더니 데이터 추출이 정상적으로 이루어졌음.

- 그래프 코드와 병합 시의 문제: 최초에 코드를 작성할 때, 표준점수는 단순히 x축에 텍스트로만 나타날 테니 굳이 정수형으로 변환하지 않아도 될 것이라고 생각하여 코드에서 따로 처리해주지 않았습니다. 그러나, drawG.py 에서는 표준 점수를 반복문을 이용해 처리하고 있었습니다. 이 때문에 문자열로 추출했던 표준 점수 데이터가 정상적으로 처리되지 않는 문제가 발생하였습니다. 문제를 인지한 이후에 표준 점수를 정수형으로 처리하도록 수정하였습니다.

```
data[line[1]][int(line[2].strip())] = [int(line[3]),int(line[4])]
```

(위의 line[2]가 표준 점수에 해당하는 부분이며, 최초에는 int() 함수를 사용하지 않았었음)

6. 결과 화면