

2024 Fall OOP Assignment Report

과제 번호 : 1-3

학번 : 20230673

이름 : 전재영

Povis ID : jason9751

명예서약 (Honor Code)

나는 이 프로그래밍 과제를 다른 소스를 인용없이 참고하거나,
다른 사람의 부적절한 도움을 받지 않고 완수하였습니다.
I completed this programming task without using any uncited sources
or receiving improper help from others.

1. 프로그램 구조 및 알고리즘

- 3번 문제는 txt파일의 데이터 베이스를 입력받아 특정 기준에 따라 정렬해 출력하는 프로그램이다.

- 프로그램은 txt 파일을 입력받아 기능을 수행한다.

1. 파일, 데이터 베이스는 각각 학번, 국어, 영어, 수학에 해당하는 데이터를 갖고 있다.
2. 프로그램에서 정렬 기준을 입력 받아, 1을 입력하면 학번, 2를 입력하면 국어, 3을 입력하면 영어, 4를 입력하면 수학, 5를 입력하면 총점을 기준으로 정렬한다. 0을 입력하면 프로그램을 종료한다.
3. 그 후, 정렬 방식을 입력 받아, 1을 입력하면 오름차순, -1을 입력하면 내림차순으로 정렬한다.
4. 정렬한 데이터 베이스를 출력한다.

- 세부 알고리즘 설명

- 파일을 열고, 데이터를 얻어와 배열에 저장한다.
- 배열의 크기는 50 * 5로, 각각 50명 학생의 학번, 국어, 영어, 수학, 총점의 데이터를 담는다.
- 학번, 국어, 영어, 수학의 데이터를 입력 받으면, 학번을 제외한 국영수의 합을 총점에 저장하고, 다음 학생의 데이터를 저장할 배열로 이동한다.
- 앞에서 저장한 데이터 배열을 이용해 정렬 기준을 입력 받는다.

- ◆ 입력이 1일 경우, 학번을 기준으로 정렬한다.
- ◆ 입력이 2일 경우, 국어 성적을 기준으로 정렬한다.
- ◆ 입력이 3일 경우, 영어 성적을 기준으로 정렬한다.
- ◆ 입력이 4일 경우, 수학 성적을 기준으로 정렬한다.
- ◆ 입력이 0일 경우, 프로그램을 종료한다.
- (정렬 기준 - 1)이 배열에서의 index를 나타내므로, 정렬 기준의 값에서 1을 뺀다.
- 이후, 정렬 방식을 입력 받는다.
 - ◆ 입력이 1일 경우, 오름차순으로 정렬한다.
 - ◆ 입력이 -1일 경우, 내림차순으로 정렬한다.
- 데이터를 앞에서 입력받은 기준과 방식으로 selection sort 알고리즘을 사용해 정렬한다.
 - ◆ Selection sort는 배열을 순회하면서 최댓값을 기록하고, 첫번째 값과 교환한다.
 - ◆ 다음 두번째 원소부터 배열을 순회하면서 최댓값을 기록하고, 두번째 값과 교환한다.
 - ◆ 이러한 방식으로 배열의 끝까지 정렬하는 알고리즘이다.
- 만약 정렬 방식이 -1일 경우 최댓값을 기록해 내림차순으로, 1일 경우 최솟값을 기록해 오름차순으로 정렬한다.
- 그후, 배열을 순회하면서 출력한다.

■ 실행 방법 및 예제

```

정렬 기준을 선택하세요 (1: 학번, 2: 국어, 3: 영어, 4: 수학, 5: 중점, 0: 종료):
3
오름차순 정렬은 1, 내림차순 정렬은 -1을 입력하세요:
-1
학번 국어 영어 수학 중점
-----
1001 85 90 78 253
1004 90 88 85 263
1003 75 85 80 240
1002 88 67 92 247
1005 88 67 92 247
정렬 기준을 선택하세요 (1: 학번, 2: 국어, 3: 영어, 4: 수학, 5: 중점, 0: 종료):
파일 편집 보기
1001 85 90 78
1002 88 67 92
1003 75 85 80
1004 90 88 85
1005 88 67 92
  
```

- 과제의 예시를 사용하면 다음과 같다.

2. 토론 및 개선 (20%)

● 토론

- 이번 과제를 통해 c++을 통한 파일 출력, 데이터 처리와 정렬 알고리즘을 작성하는

방법을 알 수 있었다.

- 아무 정렬 알고리즘을 사용해도 된다고 해서 가장 구현이 쉬운 selection sort를 사용했는데, 좀 더 효율적인 quick sort나 merge sort 등을 공부해서 구현해 봐도 좋을 것 같다.

- 개선 방향

- 파일에서 데이터를 읽어오는 과정이 직관적이지 않는데, 4개의 데이터를 읽었을 때, 앞의 학번을 제외한 3개의 데이터를 합산해 총점의 데이터를 기록한다는 것이 조금 복잡한 것 같다. 배열이 아닌 다른 데이터 형식을 사용하거나, 총점은 함수를 통해 따로 계산하는 등의 방법을 사용하는 것이 좋아 보인다.

3. 참고 문헌 (10%)

- 참고 문헌

Selection sort : https://en.wikipedia.org/wiki/Selection_sort