**2022 Spring OOP Assignment Report**

과제 번호 : ASSN1\_prob2

학번 : 20210716

이름 : 최대현

Povis ID : daehyeonchoi

**명예서약 (Honor Code)**

나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.

I completed this programming task without the improper help of others.

프로그램을 하다 보면 결정해야 할 세부 사항이 많은데, 이러한 세부 사항을 처리한 방법과 이유를 보고서에 쓰십시오.

독창적인 아이디어와 추가 기능은 보너스 점수를 받을 수 있으므로, 보고서에 명확히 기재하십시오.

문제가 여러 개인 경우, 각 문제별로 정리해서 작성합니다.

1. **프로그램 개요**

개발환경: Visual Studio 2019

Visual studio에서 프로젝트를 실행한 후 리소스 파일에 prob2\_20210716.cpp 파일을, 소스 파일에 input.txt를 드래그한 뒤 실행한다.

Input.txt: 조교님께서 제공해주시거나 스스로 만든 test case 파일이며, 행 수에는 제한이 없지만 (op, num1, num2 개행) 형태의 파일이어야 원활한 구동이 가능하다.

Output.txt: input.txt의 정보를 바탕으로 사진을 정렬하여 다시 출력하는 대상 파일

1. **프로그램의 구조 및 알고리즘**

별도의 class를 구현하지는 않았다.

사진의 정보에는 2가지 데이터 (장소, 연도)가 포함되어 있다. 이를 묶어서 p\_info라는 자료형의 구조체 변수로 만들었다. 이 때, 장소는 C++ style string으로 만들었고 연도는 int형 변수로 만들어 저장했다.

또한, 데이터를 정렬할 때 처음 input.txt에서 받아온 데이터의 개수가 변하지 않으므로 구현의 편의성을 위해 연결리스트가 아닌 배열로 구현하였다.

텍스트이(가) 표시된 사진

자동 생성된 설명

C++ 스타일로 fin fout 객체를 만들고 입출력 파일을 연 뒤, P\_info type의 변수로 구성된 iu\_photo 배열을 만들었다. (이 때 배열의 size는 100으로 구현했지만, 필요에 따라 코드 수정을 통해 size를 늘릴 수 있음)

텍스트이(가) 표시된 사진

자동 생성된 설명

그 뒤 main 함수를 간단히 하기 위해 선택정렬 함수 sel\_sort를 따로 구현하였다. 그 뒤 마지막으로 배열 iu\_photo를 0번 인덱스부터 차례대로 출력하고 파일을 닫는 것으로 main함수의 호출이 종료된다.

텍스트, 스크린샷, 화면, 검은색이(가) 표시된 사진

자동 생성된 설명

**함수 설명**

* **void** sel\_sort(**p\_info** iu\_photo[], **int** photo\_num): 선택 정렬 방식으로 iu\_photo 배열에 있는 데이터들을 정렬해주는 함수이다. 문제의 규칙에 맞게 연도를 우선순위로 정렬하였고, 연도가 같은 경우 장소의 이름을 알파벳순으로 오름차순 정렬하였다.

텍스트이(가) 표시된 사진

자동 생성된 설명

보편적인 선택 정렬 방식으로 min\_idx값을 계속해서 갱신해나가고, current index에 저장된 값과 바꾸며 current index를 하나씩 오른쪽으로 옮기는 방식을 채택했다. 이 때 연도값은 swap함수로 바꾸었고, place는 sort함수 안에서 정렬했다.

* **void** swap(**int \***a, **int \***b): 일반적인 call-by-address 방식의 swap함수이다.

텍스트이(가) 표시된 사진

자동 생성된 설명

1. **토론 및 개선**

Selection sort: 우리말로 선택 정렬이라고 하며, 정렬되지 않은 데이터에 대해 가장 작은 데이터를 찾은 뒤 현재 데이터와 교환하는 방식이다.

**단점:**

1. 데이터를 하나씩 비교하기 때문에 시간복잡도 측면에서 효율적이라고 할 수는 없다.

2. 원소 교환 횟수가 적어서 교환 횟수가 많은 정렬 케이스에서는 유리하지만, 반대로 데이터 개수가 많아지고 교환 횟수가 적은 정렬 케이스에서는 불리하다.

위 프로그램은 깔끔하게 구동되지만 아직은 절차지향적에 머물러 있다는 느낌이 든다. 아직 객체 지향 프로그래밍에 익숙하지 않아서 class 등의 문법을 적극적으로 활용하지 못했는데, 기회가 되면 객체지향적으로 코딩을 다시 해보고 싶다.

1. **참고 문헌**

선택 정렬 – Wikipedia

[https://ko.wikipedia.org/wiki/%EC%84%A0%ED%83%9D\_%EC%A0%95%EB%A0%AC](https://ko.wikipedia.org/wiki/%EC%84%A0%ED%83%9D_%EC%A0%95%EB%A0%AC%20)

(개념 부분을 참고하였으며 C 코드를 모방하진 않았음)