**2022 Spring OOP Assignment Report**

과제 번호 : 3

학번 : 20210054

이름 : 정하우

Povis ID : howru0321

**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.

각 문항별 설명은 편의를 위한 것으로, 삭제하고 제출한다.

1. **프로그램 개요**
   * Input.txt에 있는 미로에서 열쇠를 찾아 출구를 가는데 까지 걸리는 최단경로를 찾는 문제이다. 열쇠는 k, 출구는 e 통행가능지역은 0, 통행불가능지역은 1로 나타낸다.
   * DFS(Depth First Search)알고리즘을 통해 해결하였다
2. **프로그램의 구조 및 알고리즘**

* 모든 경로를 다 지나간다. 어떠한 지점에서 4방향 중 갈 수 있는 방향 두 가지 이상일 경우 위 왼 아래 오 순으로 경로를 간다. 더 이상 갈 수 없는 경우는 다시 돌아와 다시 위 왼 아래 오 순으로 비교하여 그 경로를 간다. 이러한 반복은 사용자 정의 함수를 재귀함수로 나타내면서 구현하였다.
* 처음에 열쇠의 위치와 출구의 위치를 파악한 후, 시작점과 열쇠사이의 최단거리, 열솨와 출구 사이의 최단거리를 구하고 이를 더하여 output.txt에 저장하였다
  + - i, j, k, l : for문에서 사용되는 int형 변수
    - map[10][10] : input.txt에 있는 정보를 담아오는 2차원 배열
    - int strt(char map[][10], int spox, int spoy, int fpox, int fpoy, int dis, int l) : 최단경로를 찾는 함수
    - sopy : 시작점의 y좌표
    - sopx : 시적점의 x좌표
    - fpoy : 끝점의 y좌표
    - fpox : 끝점의 x좌표
    - dis : 최종적으로 최단경로가 담기는 변수
    - l : 어떤 경로의 경로 길이가 담기는 변수
    - kdis : 출발점과 열쇠 사이의 최단경로
    - kedis : 열쇠와 출구 사이의 최단경로

1. **토론 및 개선**
   * + DFS라는 알고리즘을 처음 알게되었다
     + 처음 봤을 때 생각해 낸 알고리즘이 DFS와 매우 유사하였다. 그런데 이를 구현하는 과정에서 재귀함수를 사용한다는 생각을 하는데 너무 오랜시간이 걸렸고, 이 생각을 하기 전까지는 진전이 거의 없었다. 특히 전역변수를 사용하지 못해 프로그램을 완성하는데 어려움이 있었다. 물론 재귀함수를 이 프로그래밍에 적용시킬때도 머리가 복잡해 프로그램을 완성하는데 시간이 꽤나 걸렸다. 재귀함수와 같이 복잡한 알고리즘에도 익숙해 져야겠다는 생각이 들었다.
     + 전역변수를 사용하면 조금 더 간결하게 알고리즘을 만들 수 있을거 같다
     + BFS(Breadth First Search)알고리즘을 통해서도 이 문제를 해결할 수 있다는 사실을 알게 되었다. 다음에는 이 알고리즘을 통해 구현해 보고 싶다.
2. **참고 문헌**

* <https://blog.hexabrain.net/269>

-DFS알고리즘의 개념를 참고함