

과제 개요 #2: 지하철 최단경로/최소환승

1. 개요

다익스트라 알고리즘은 그래프 상에서 출발점과 도착점 사이의 최단 거리를 구하는 알고리즘이다. 다익스트라 알고리즘은 단순히 이론적인 알고리즘이 아니라 실생활에 많이 응용되고 있는 알고리즘으로, 대표적으로 네트워크상에서 통신 가능한 최단 경로를 구하는 라우팅에 많이 쓰이고 있으며, 또한 자동차 네비게이션이나 지금부터 우리가 구현해야 할 지하철 최단 경로에도 대표적으로 쓰이고 있다. 이러한 알고리즘을 실생활의 문제에 적용하여 해결함으로써 프로그래밍 능력을 향상하고 응용력을 기를 수 있다.

2. 목표

그래프를 구현 및 설계하고 그래프에 최단거리 알고리즘을 적용함으로써 자료구조 설계능력과 프로그래밍 능력을 향상시킨다.

3. 팀 구성

개인별 프로젝트

4. 개발환경

- 1) OS: Ubuntu 20.04
- 2) Tools: vim, vs code, gcc

5. 보고서 제출 방법

- 1) 제출할 파일
 - **보고서(.pdf)와 소스파일을 함께 압축하여 제출**
 - 압축파일명: [2]_학번_ver00.tar.gz
- 2) 제출할 곳
 - 온라인(e러닝) 제출

6. 보고서 양식

- | |
|---|
| <ol style="list-style-type: none">1. 과제 개요2. 설계3. 구현<ul style="list-style-type: none">- 각 함수별 기능4. 테스트 및 결과<ul style="list-style-type: none">- 테스트 프로그램의 실행 결과를 분석 (캡처화면과 함께)5. 소스코드 (주석포함) |
|---|

7. 설계 및 구현

1) 구현 내용

사용자로부터 출발역과 도착역을 입력받고, 옵션으로 최단 경로와 최소 환승을 선택한다. 사용자가 입력한 정보를 바탕으로 역 간 거리 정보가 저장된 텍스트파일을 읽어 인접행렬을 만든 뒤 다익스트라 알고리즘을 이용하여 최단경로로 목적지까지 도착하는 경로를 출력한다.

2) 세부 기능 및 기능별 요구 조건

① 역 정보가 저장된 텍스트파일을 읽어 인접행렬을 생성하는 함수 구현

- 역 간 소요시간 파일을 호선별로 제공함

파일명: 1호선.csv, 2호선.csv, ...

파일형식: 콤마(,)로 구분되고 A역,B역,(A->B)소요시간 순서로 제공

- 환승역 파일 제공함

파일명: 환승정보.csv

파일형식: 콤마(,)로 구분되고 A역,B역,(A->B)소요시간 순서로 제공

- 역 번호에 대한 역 이름 정보파일 제공함

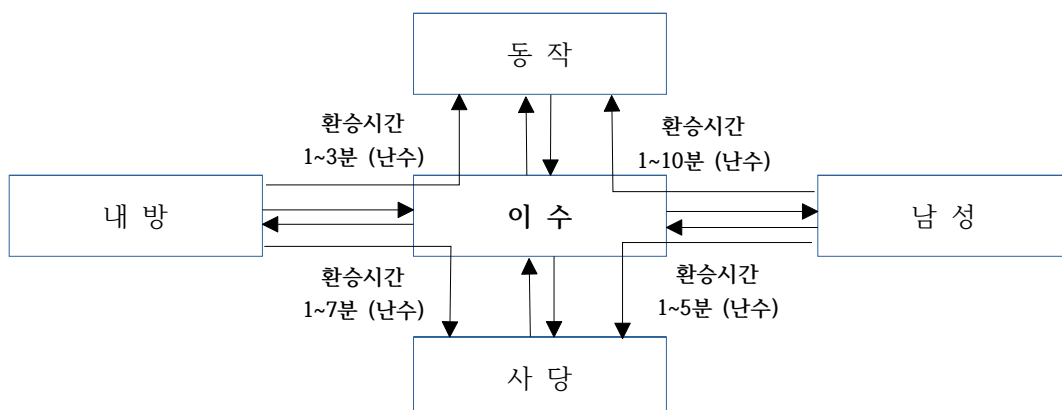
파일명: 역이름.csv

파일형식: 콤마(,)로 구분되고 역 번호, 역 이름 순서로 제공

- 제공된 파일을 각각 읽어 하나의 인접행렬을 2차원 배열로 생성
총 역의 개수의 2차원 배열을 동적으로 생성해야 함

② 환승할 경우 환승정보 파일에서 제공한 최대 환승 소요시간 안에서 난수 값을 생성하는 함수구현

- ex) 7호선과 4호선이 지나는 이수역의 경우



③ 다익스트라 알고리즘을 이용하여 최단경로 탐색함수 구현

- 생성한 인접행렬을 이용하여 출발역부터 목적지까지 최단경로의 모든 역을 출력

④ 사용자로부터 최단경로와 최소환승의 옵션을 입력받아 각각의 조건에 맞게 수행

- 출발역과 도착역을 먼저 지정한 뒤 옵션을 받아야 함

ex) 출발역: 과천

도착역: 인천대입구

1. 최단경로, 2. 최소환승: 1

⑤ 기본적인 예외 처리 함수 구현

- 잘못된 역 이름을 입력받은 경우 오류내용을 출력하고 재입력 받도록 처리
- 출발역과 도착역이 같은 경우 오류내용을 출력하고 재입력 받도록 처리

3) 프로그램 실행 예

출발역을 입력해주세요: 과천

도착역을 입력해주세요: 인천대입구

방식? 1. 최단경로 2. 최소환승

:1

최단경로

<출발>

-><4호선> 과천

-><4호선> 대공원

-><4호선> 경마공원

-><4호선> 선바위

-><4호선> 남태령

-><환승 : 소요시간 4 분> 사당

-><2호선> 낙성대

...

...

-><인천1선> 캠퍼스타운

-><인천1선> 테크노파크

-><인천1선> 지식정보단지

-><인천1선> 인천대입구 <도착>

소요시간 : 97 (84 + 환승 소요시간 : 13) 분

정거장 수 : 42 개

4) 설계 구성 요소

- * 목표 설정

- 주어진 요구 조건을 이해하고 명확한 설계 목표를 설정한다.
- 설계의 목표 및 요구조건을 문서화한다.

- * 분석

- 목표 설정에서 명시한 요구 조건을 분석하고 해결을 위한 기본 전략을 수립한다.
- 문제 해결을 위한 배경 지식을 이론 강의에서 듣고 개별적으로 학생들이 다양한 경로를 통해 자료를 찾아 분석한다.

- * 합성 (자료구조 설계)

- 분석 결과를 토대로 적절한 자료구조를 도출한다.
- 도출된 자료구조를 토대로 모듈을 작성한다.

- * 제작 (구현)

- 각 모듈의 입출력을 명확하게 기술한다.
- 자료구조와 모듈을 C언어로 구현하고 이를 컴파일하여 실행 파일을 만든다.

- * 시험 및 평가 (성능평가)

- 모듈의 입출력 명세에 따라, 다양한 입력 데이터를 작성하고 모듈을 테스트한다.
- 작성된 실행 파일이 안정적으로 수행되는지 다양하게 테스트하고 이를 평가한다.

- * 결과 도출

- 안정적으로 수행되는 최종 결과(Output)을 캡처하여 최종 결과물은 보고서에 반영한다.

8. 평가 도구

- 1) 실행 보고서 평가
- 2) 실행 평가

9. 평가 방법

- 1) 평가 도구 1) 에 대한 평가 방법

- * 소스 코드 분석 및 새로운 모듈의 설계가 제대로 이루어졌는가?

- 설계 요구 사항을 제대로 분석하였는가?
- 설계의 제약 조건을 제대로 반영하였는가?
- 자료구조가 적절히 설계되었는가?
- 모듈 설계 방법이 적절한가?

- * 문서화

- 소스 코드에 주석을 제대로 달았는가?
- 설계 보고서가 잘 조직화하고, 잘 쓰였는가?

- 2) 평가 도구 2) 에 대한 평가 방법

- * 요구조건에 따라 올바르게 수행되는가?

10. 기타

- 1) 지연 제출 시 감점: 1일 지연 시 마다 30% 감점, 3일 지연 후 부터는 미제출 처리
- 2) 압축 오류 or 파일 누락: 50% 감점 처리
- 3) copy 발견시 F 처리