

# 프로젝트

## 1. 라인에디터 (Line Editor) 프로젝트

### 목표:

- 선형 리스트를 사용하여 텍스트 에디터 구현

### 제한사항:

- 라인 단위로 입력 또는 삭제를 할 수 있음
- 고정된 수의 명령어를 입력받아 동작하며 커서를 사용하지 않음
- Buffer 라는 연결리스트를 사용하여 사용자가 입력하는 텍스트를 저장

### 동작:

- 삽입: 사용자로부터 라인의 번호와 라인의 텍스트를 입력받아 Buffer 연결리스트의 지정된 위치에 삽입
- 삭제: 사용자로부터 라인의 번호를 입력받아 지정된 라인의 텍스트를 Buffer 연결 리스트에서 삭제
- 'w': Buffer 연결리스트의 텍스트를 파일에 저장 (파일이름을 사용자로부터 입력받아 저장) - 파일이 없으면 읽지 못한다고 메시지 출력
- 'r': 파일에 저장된 문서를 읽어 Buffer 연결리스트로 구성하여 문서 편집 준비 (파일이름을 사용자로부터 입력받아 파일을 읽어옴) - 파일이 없으면 읽지 못한다고 메시지 출력
- 'd': 사용자가 지정한 라인을 Buffer 연결리스트에서 삭제
- 'i': 사용자가 입력한 텍스트를 Buffer 연결리스트의 지정된 라인에 삽입
- 'l': Buffer 연결리스트로 구성된 문서 내용 모두를 화면에 출력
- 'f': Buffer 연결리스트에서 주어진 문자열이 포함된 줄의 텍스트를 출력
- 'q': 텍스트 에디터를 종료 (편집한 문서가 수정이 되었다면 저장할 것인지 사용자에게 확인하고 저장할 수 있도록 함)

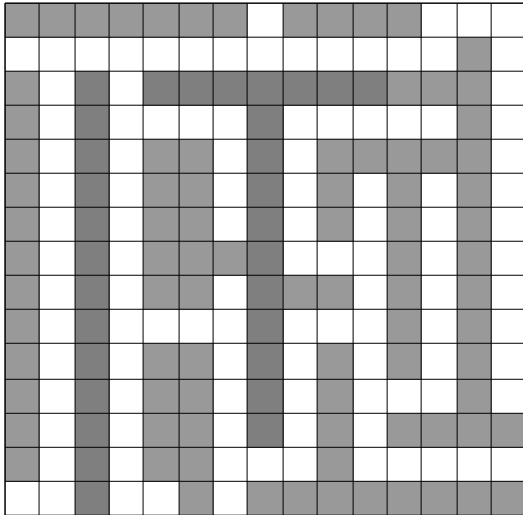
## 2. 미로 탐색 프로젝트

### 목표:

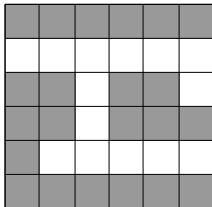
- 스택 사용하여 미로 탐색 기능 구현

### 제한사항:

15 x 15 의 미로를 생성



6 x 6 미로 생성



위의 좌측이 2차원 배열의 (0, 0)에 해당

입구: (1, 0), 출구 (14, 15), (4, 5)

### 동작:

두 개의 미로를 2차원 배열로 구성하고 각 미로의 (1, 0)에서부터 시작해서 출구까지 나가는 경로를 찾아내는 프로그램 작성하고 찾은 경로를 2차원 미로에 표현하시오

### 3. 이진탐색트리를 활용한 영어사전 프로젝트

#### 목표:

- 이진탐색트리를 이용하여 영어사전 구현하기

#### 제약조건:

- 영어 알파벳 순으로 정렬되며 의미도 함께 저장

#### 동작:

- 'i': 입력 모드 - 사용자가 단어와 의미를 입력하여 이진탐색트리에 노드를 추가
- 'd': 삭제 모드 - 사용자 입력한 사전을 삭제하는 것
- 's': 탐색 모드 - 사용자로부터 영어 단어를 입력받아 의미를 화면에 출력
- 'p': 출력 모드 - 모든 단어와 의미를 화면에 출력
- 'w': 저장 모드 - 단어의 내용을 파일로 저장 (프로그래머가 지정한 파일로)
- 'q': 종료 모드 - 단어장 애플리케이션 종료

## 4. 이산 이벤트 시뮬레이션 프로젝트

이산 이벤트 시뮬레이션은 모든 시간의 진행이 이벤트의 발생에 의해 이루어진다. 이벤트가 발생하면 시간이 진행된다. 많은 이벤트가 발생할 경우 이 이벤트들은 시간적으로 먼저 발생한 이벤트가 먼저처리 되어야 한다. 이를 위해 우선순위 큐를 이용하여 이벤트를 저장하고 이벤트의 발생 시각을 우선순위로 하여 이벤트를 처리하는 시뮬레이션 개발

### 목표:

- 최소 힙(heap)를 이용한 아이스크림 가게 시뮬레이션 프로그램 구현
- 아이스크림 가게에 손님이 들어오고 나가는 과정을 시뮬레이션 하는 것
- 몇 개의 의자를 비치해야 이익을 극대화할 수 있는지 시뮬레이션

### 제약조건:

- 아이스크림 가게의 의자가 파러미터
- 손님이 도착했을 때 의자가 없으면 그냥 나간다.
- 아이스크림 가게에서 발생하는 이벤트
  - 손님이 도착하는 이벤트 (Arrival)
  - 손님이 주문하는 이벤트 (Order)
  - 손님이 가게를 떠나는 이벤트 (Leave)

각각의 이벤트들은 이벤트가 발생한 시각과 소님을 가짐  
도착한 손님은 랜덤한 시간 후에 주문 이벤트를 발생시킴  
손님은 주문을 하고 나서 랜덤한 시간 후에 가게를 떠나는 이벤트를 발생

### 동작:

- 시뮬레이션 처음부분에서 일정 시간 동안 손님이 도착하는 이벤트를 몇 개 발생시켜 우선순위 큐에 저장 - 우선순위 큐는 이벤트가 발생한 시각 순으로 이벤트를 꺼내어 처리하고 우선순위 큐에 더 이상 이벤트가 없으면 프로그램을 종료
- 이벤트는 각각의 종류에 따라 다르게 처리한다.
- 손님이 도착하는 이벤트 처리: 먼저 손님의 숫자와 현재 남아 있는 의자를 비교하여 남아 있는 의자가 더 많으면 손님을 받고 남아 있는 의자의 수는 손님의 숫자만큼 줄어든다. 만약 의자가 더 적으면 손님은 주문을 하지 않고 나간다.
- 주문 이벤트 처리: 손님의 숫자대로 주문을 받고 잠시 후에 손님들이 떠나는 이벤트를 발생시킨다.
- 손님이 떠나는 이벤트: 떠나는 숫자만큼 남아 있는 의자의 개수를 증가

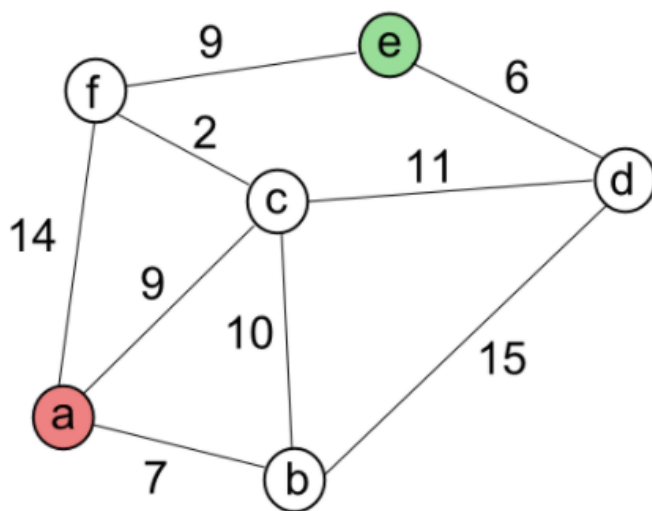
손님들이 가게에 들어와서 항상 일정한 시간 후에 주문을 하거나 일정 시간 후에 가게를 나가는 것은 의미가 없으므로 손님이 도착하는 시간, 손님의 숫자, 주문하는데 걸리는 시간, 가게에 머물러 있는 시간은 모두 랜덤하게 발생 - <세 개의 이벤트는 하나의 우선순위 큐에서 관리되면 시간이 앞선 것을 먼저처리 한다.>

## 5. 다익스트라(dijkstra) 알고리즘을 활용하여 우체국의 우편물 포워딩 테이블 구성 프로젝트

### 목적:

- n 개의 우체국으로 구성된 무방향 가중치 그래프서 특정 목적지로 가는 우편물이 접수되었을 때 인접 우체국 중 어느 우체국으로 우편물을 전달해야 하는지 알려주는 테이블 구성

제약조건: - 그래프의 예



- 인접 리스트/ 인접행렬 중 하나를 선택하여 그래프를 표현
- 임의의 그래프에 대해서 검증할 수 있는 기능 구현

### 동작:

- 각 우체국 마다 목적지 우체국까지 우편물을 보내기 위해 인접한 우체국에 대한 정보를 가지고 있어야 함
- 'l': 출발지 우체국과 목적지 우체국을 입력받아 우편물을 전달할 다음 우체국을 화면이 출력
- 'p': 모든 우체국에서 각 목적지까지 우편물을 보내고자 할 때 전달해야할 다음 우체국 목록 모두를 화면에 출력
- 'q': 프로그램 종료