

목표

본 과제 수행을 통하여 Class 와 Method 등을 이용한 Object Oriented programming 을 수행함으로써 객체지향 및 Python 에 대한 이해도와 프로그래밍 능력을 기를 수 있음.

과제 개요

건물에 설치되어 있는 엘리베이터와, 이용객 등을 프로그래밍으로 구현하고, 해당 엘리베이터가 동작하는 알고리즘을 본인이 작성하여, 그 알고리즘이 얼마나 효율적으로 움직이는지를 테스트 함.

실험 전제조건

- 모든 건물은 최소 2 층 이상의 건물이며, 지하는 사용하지 않습니다.
- 실험시, 이용객은 1 명 이상이어야 합니다.
- 실험을 시작할 때, 엘리베이터는 1 층에서 시작합니다.
- 실험할 때, 이용객은 언제나 건물의 층 범위 내에서 움직입니다.
(Ex. 6 층 건물에서 8 층 출발 / 도착하려 하는 이용객과 실험을 진행할 수 없습니다.)
- 이용객은 엘리베이터를 통해서만 층을 오르거나 내릴 수 있습니다.
- 이용객은 언제나 현재 층과 다른 층으로만 이동합니다.
(Ex. 3 층에서 3 층으로 이동하는 경우 없음. 올라가거나 내려가야 함.)
- 이용객은 본인이 내려야 하는 층에서만 내려야 함.
(Ex. 4 층에서 2 층으로 이동하는 이용객이 3 층에서 내렸다 다시 타는 경우 없음)
- 엘리베이터에는 수용 인원이 한정되어 있으며, 그 이상 이용객이 탈 수 없습니다.
(Ex. 5 명 정원인 엘리베이터에서 누군가 내리기 전에는 탈 수 없습니다.)
- 이용객은 한 번 엘리베이터에서 타고 내리면, 더 이상 엘리베이터를 타지 않습니다.
(Ex. 3 층 -> 1 층 이용객일 경우, 1 층에서 다시 타는 일은 없습니다.)
(Ex. 3 명의 이용객으로 실험하고, 3 명 모두 원하는 층에서 내리면, 실험이 종료됩니다.)
- 실험에 참여하는 모든 이용객이 엘리베이터를 이용하여 자신이 이동하고자 하는 층에 이동하면 실험이 종료되며, 엘리베이터는 굳이 다시 1 층으로 돌아오지 않아도 됩니다.
- 엘리베이터가 이동할 때마다 몇 층을 이동했는지 기록하며, 실험이 종료되면 총 몇 층을 이동했는지 기록하여, 얼마나 효율적으로 엘리베이터가 이동했는지 확인합니다.
(Ex. 1->4->2->3 으로 이동할 경우, $3+2+1 = 6$ 개 층을 이동하였음.)

구현을 위한 Class 별 요약

- class Customer : 이용객
 - 엘리베이터를 타는 층 (이하 StartFloor), 내리는 층 (이하 DestFloor)을 저장하고 있으며, 이 때, StartFloor 와 DestFloor 는 언제나 달라야 합니다.
- class Elevator : 엘리베이터
 - 수용인원 (이하 Capacity), 탑승이용객 (이하 Customers), 최고층 (이하 MaxFloor) 저장하고 있으며, 탑승이용객의 숫자는 수용인원을 넘을 수 없습니다.
 - 엘리베이터는 탑승 (이하 Load), 이동 (이하 Move), 하차 (이하 Unload)를 수행 할 수 있으며, 각 행동의 인자는 본인이 원하는 대로 작성하시면 됩니다.
- class Building : 건물
 - 엘리베이터와 실험이용객들을 저장하고 있으며, 엘리베이터의 MaxFloor 는 건물의 층 높이와 같습니다.
 - 실험 이용객들의 StartFloor 나 DestFloor 가 건물의 MaxFloor 를 넘는 경우, 실험을 진행할 수 없습니다.
 - 실험하기(이하 Run)를 할 수 있으며, 실험이용객들이 모두 원하는 대로 이동하면 실험을 종료하며, 결과로 엘리베이터가 총 몇개 층을 이동했는지 출력합니다.
 - 엘리베이터가 각 손님들의 요청에 따라 어떻게 이동할 지 결정하는 알고리즘 부분은 이 부분에 작성하는 것을 추천합니다.
 - 실험 중, 엘리베이터가 어떻게 이동했는지, 현재 몇 명의 손님이 엘리베이터에 탑승 중인지, 내렸는지, 타려고 기다리고 있는지 등을 중간에 출력하는 것을 추천합니다.

테스트 프로그램 및 시나리오:

- 본 과제 공지와 함께 나간 테스트 프로그램은 실행하면 3 가지 값을 입력받습니다.
 - 건물 및 엘리베이터의 층 높이 (2 이상이어야만 함)
 - 실험에 참여하는 이용객 수 (1 이상이어야 함)
 - 엘리베이터의 수용인원 (1 이상이어야 함)

위의 입력 조건이 잘못 되었을 경우, 다시 입력을 요청하도록 하여 정상적인 실험 조건이 요청되면, 입력한 값에 맞는 엘리베이터, 건물과 이용객들을 생성합니다. 이때 이용객의 StartFloor 와 DestFloor 는 건물의 층 범위 내에서 무작위로 결정됩니다. 위 입력 조건에 맞는 건물, 엘리베이터, 이용객들이 생성되면, 실험을 시작합니다.

- 테스트 시나리오

위 전제조건에 맞게 잘 짰는지 (잘못된 실험 조건이 들어 올 경우, 실험을 진행하지 않고 무엇이 잘못 되었는지 알려줌), 알고리즘이 얼마나 효율적인지 (엘리베이터가 최대한 덜 움직이고 실험을 종료할 수록 효율적임) 확인하기 위한 시나리오입니다.

- 테스트 시나리오 1

조건 :

4 층 건물 / 엘리베이터 수용인원 2 명

실험인원 3 명: (2->5), (1->2), (6->3)

예상 결과 :

실험이 불가능하다고 경고해야 함.

특히 어떤 이용객의 요청때문에 불가능한지 설명할 수록 좋음.

위 조건에서는 (2->5), (6->3)으로

- 테스트 시나리오 2

조건 :

5 층 건물 / 엘리베이터 수용인원 2 명

실험인원 3 명 : (1->4), (2->4), (3->1)

예상 결과 :

6 개 층 이동이 가장 효율적임. (1->2->4->3->1)

○ 테스트 시나리오 3

조건 :

5 층 건물 / 엘리베이터 수용인원 2 명

실험 인원 : 4 명 : (1->4), (2->4), (3->4), (3->1)

예상 결과 :

8 개 층 이동이 가장 효율적임. (1->2->4->3->4->1)

○ 테스트 시나리오 4

조건 :

테스트 프로그램과 같음.

건물 층 높이, 수용인원, 실험인원을 입력받음.

각 이용객의 StartFloor 와 DestFloor 는 건물의 층 범위 내 무작위

예상 결과 :

정상적인 입력 조건 하에 어떤 값을 입력하더라도 실험이 정상적으로 종료 가능하다는 것을 보이면 됨.

예제로 주어진 프로그램에서는 엘리베이터가 매 층마다 멈추며 각 층에서 타야 하는 이용객들을 최대한 태우고, 내려야 하는 이용객들을 내린 다음, 1 개 층씩 이동하고, 마지막 층(1 층이나 꼭대기 층)에서는 반대방향으로 이동하는 방식으로 동작합니다. 그러므로 본 테스트 시나리오의 예상 결과처럼 효율적이지 않을 수도 있습니다.

평가 기준

본 문제 해결을 위한 프로그램을 객체지향적으로 잘 작성하였는가?

작성한 Class 가 적합한가, 필요한 method 만 존재하는가?

엘리베이터가 얼마나 효율적으로 동작하는가?

주의 사항

개인별 과제이며 팀으로 수행할 수 없음

모든 Class, method 에 대하여 보고서를 작성하여, 본인이 작성한 코드가 무슨 의미를 가지고 있는지 설명하여야 함.

다른 학생과 코드 공유 금지. (카피 적발시 원본 / 사본 모두 0 점 처리)

전역변수를 사용하지 말 것

힌트

1. 처음에 사용자로부터 건물의 층 수, 고객 수, 엘리베이터 수용인원 등을 물어본 뒤 (혹은 시나리오 조건에 따라 설정 한 뒤) 이를 Building 인스턴스 생성하는데 사용하면 됨.
2. 그 후, 실험을 수행하는 함수에서 모든 손님이 다 원하는 층에 도착할 때 까지 엘리베이터를 적정한 알고리즘을 통해 이동, 출력시키면 됨
3. 무작위의 숫자 생성은 `random.randint(a, b)`를 이용하면, a 에서 b 사이의 임의의 숫자를 사용할 수 있습니다. (사용하기 전에 `import random` 을 해 놓아야 합니다.)
4. 문제 내용이 이해되지 않을 경우 샘플로 제공한 실행파일을 실행해 볼 것

추가점수

Inheritance 를 사용하여 Class 를 작성하면 가산점을 부여 할 예정

과제 제출 및 마감

6 월 27 일(토) 24:00 까지 제출

1. 과제 내용에 대한 python code file
2. 본인의 코드에 대한 설명 보고서, 결과물 등

ASLKAU@gmail.com 으로 코드와 보고서 파일 이름을 학번으로 첨부하여 보낼 것

Ex. 메일 제목 : [프로그래밍 기초] 텀프로젝트_2015122000

첨부파일 이름 : 2015125000.py (코드) / 2015125000.pdf (보고서)

문의 사항

JS.Yoo.5B@gmail.com 으로 메일을 보내거나, 010-3129-8354 로 연락

과제 수행 중 모르는 것에 대해서는 질문 받지 않으며, 본 공지에서 설명하지 않은 내용이 있는 경우에만 대답할 것임.