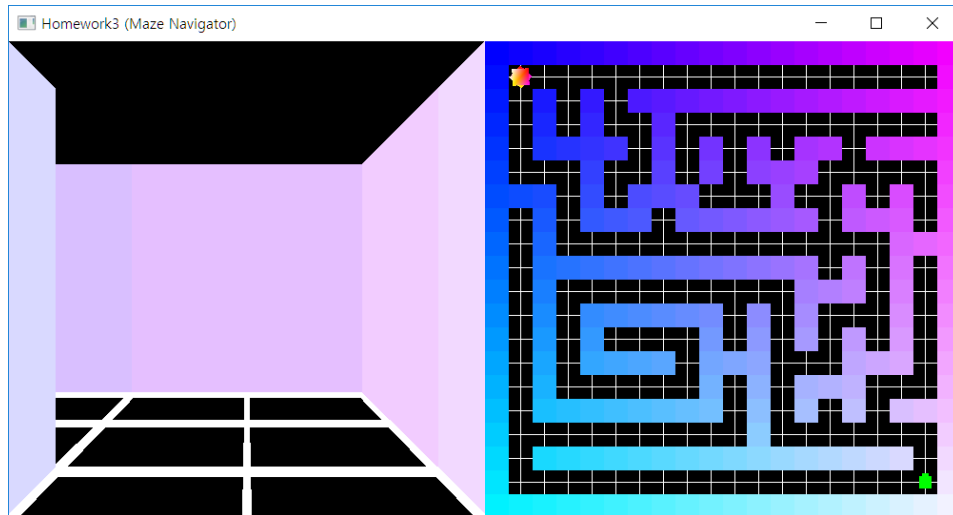


# Computer Graphics Homework #3

## Maze Navigator 만들기:



주어진 Starting Code를 고쳐서 다음과 같은 조건을 만족하는 프로그램으로 완성하라.

1. 'A' 키와 'D' 키의 동작을 변경하여 'A'키를 누르면 카메라가 왼쪽으로 회전, 'D'키를 누르면 카메라가 오른쪽으로 회전함 (왼쪽 화면의 시점 변화뿐만 아니라 오른쪽 화면의 녹색 카메라 모습 또한 회전된 모습으로 그려져야 함) (배점: 25점)
2. 카메라가 벽을 뚫고 들어가지 못하게 벽 근처에서 움직임을 제한함 (배점: 25점)
3. 'Q' 키를 누르면 목표점까지 자동으로 가장 빠른 길을 찾고 찾은 길을 빨간 선으로 보여줌 (배점: 20점)
4. 'SpaceBar'를 누르면 카메라가 찾은 길을 따라 움직임 (이 때 카메라의 회전 등이 갑작스럽게 되지 않고 부드럽게 돌아감) (배점: 20점)
5. 보고서 (배점: 10점)

힌트:

- 길을 찾는 알고리즘으로 A\* (a-star) Search Algorithm을 찾아보고 구현해 볼 것  
(보고서에 A-star 알고리즘에 대해 설명하고, 내용을 이해하여 직접 구현할 것)
- 찾은 길을 그리기 위한 빨간 선은 주어진 cube.h를 이용하여 빨간 얇고 긴 사각형을 그리  
는 방식으로 할 것 (Grid 그리는 코드 부분을 보면 도움이 될 것임)
- 주어진 코드는 마음대로 고쳐도 되나, 반드시 main.cpp만 고치며, 제출 시 main.cpp만 제출  
할 것 (하나의 파일로 작성)
- 주어진 코드에서는 연속적인 Keyboard/mouse 입력을 받기 위해 GetAsyncKeyState(..) 함수  
를 사용하였으며, 관심 있는 사람들은 왜/어떻게 사용하는지 찾아볼 것

제출물:

- Code 및 보고서 제출: blackboard.sejong.ac.kr
- 다음주 월요일 자정 (11월 21일 23:59)까지
- source code(main.cpp) 및 보고서 + screenshot 1개를 zip으로 압축하여 제출
- 보고서는 A\*알고리즘 및 시행착오와 해결의 아이디어 등을 기록할 것.

더 생각해 볼 문제:

숙제로 할 필요는 없으나 다음과 같은 문제들에 대해서도 고민해볼 것

1. 자동으로 미로를 만들어내는 알고리즘
2. 부드러운 최단 거리 길을 만들어내는 방법  
(직선이 아닌 부드러운 곡선으로 된 길을 찾아냄)