

Computer Graphics - Assignment #2 (15점)

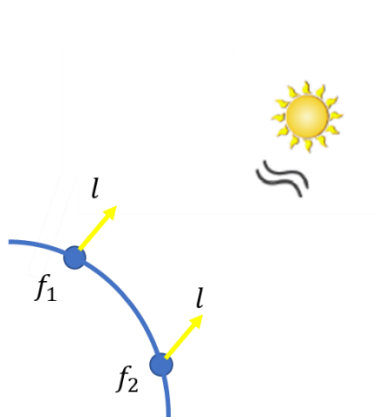
1. 사이버 캠퍼스를 통하여 제출하십시오.
 - ① 사이버 캠퍼스의 컴퓨터 그래픽스 강의에 접속하시어 과제 탭에서, 과제 2에 파일을 첨부하여 제출하시면 됩니다.
 - ② **제출 파일은 js 파일 하나**입니다. HTML과 클래스 JS파일들은 제공된 것을 그대로 사용하시면 됩니다.
 - ③ lightFragment.js파일을 구현한 뒤, **제출 전에 파일 이름을 “이름_학번.js”로 바꾸어 제출해 주십시오.** (ex, “김형기_202012345.js”)
 - ④ 제출 파일은 표절 검사 프로그램을 거칩니다. 표절로 판단될 경우 0점 처리합니다.
2. 제출 기한: **12월 06일 23:55분까지**
 - ① **추가 제출(늦은 제출)은 받지 않습니다.** 기한 내에 제출해 주십시오.
 - ② 업로드 시간까지 고려하시어 여유 있게 제출해주십시오.
3. 기타 문의사항은 이메일(hk.kim@cnu.ac.kr)로 연락 바랍니다.

채점 기준

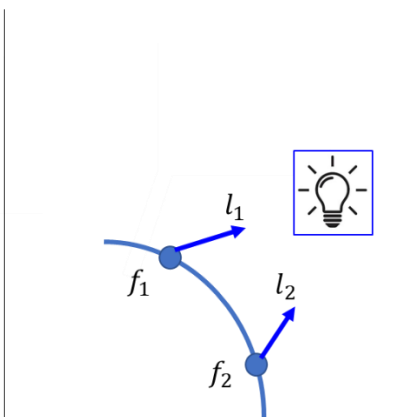
- ① **다중 점 광원 효과 구현 (15점)**
 - ◆ **올바로 동작하지 않는 경우 구현 정도에 따라 부분점수 부여**
 - ◆ **감점 예시**
 - 조명의 위치가 올바르게 정의되지 않거나, 위치 변경이 제대로 되지 않음(-2)
 - **Attenuation**이 올바르게 동작하지 않음(-2)
 - **HTML 등 다른 파일을 수정함(-5)**

<lightFragment shader 구현>

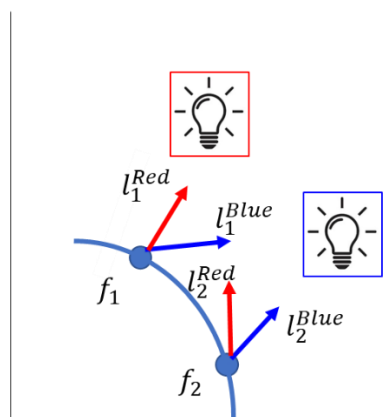
- Directional Light에 대해 배운 내용을 바탕으로 다중 Point Light 효과를 계산할 수 있는 프래그먼트 Shader를 구현하시오.
- **첨부한 Assign2 Start File의 PointLight.js, main.js의 소스 및 주석을 참고**하시오.
 - 과제와 관련된 코드에는 [Assign2]로 주석이 달려 있음
- Point Light로 인한 물체의 라이팅 효과 계산 방법은 아래와 같다.
 - 임의의 2개 fragment f_1, f_2 에 대해서, Directional Light의 경우 아래 왼쪽 그림과 같이 빛을 향하는 벡터 l 이 fragment의 위치와 관계없이 동일한 방향을 가진 벡터로 가정하여 계산하였다.
 - Point Light의 경우 아래 가운데 그림과 같이 가까이 존재하는 광원이기 때문에, fragment별로 빛과 마주하는 벡터 l 을 다르게 하여 라이팅 효과를 계산한다. 그 결과, fragment별로 normal과 l 이 이루는 각도가 빠르게 바뀌기 때문에 빛의 효과가 뚜렷하게 나타나게 된다.
 - 여러 개의 Point Light가 존재하는 경우 아래 오른쪽 그림과 같이, fragment별로 개별적인 점 광원에 따른 색상을 계산 후에 모두 더하는 방식으로 최종적인 픽셀의 색상을 결정한다.



Directional Light의 경우



Point Light의 경우



다중 Point Light의 경우

- Point Light의 경우 근접 광원이기 때문에, 거리에 비례하여 빛이 감쇄 (attenuation)하는 효과를 고려하여 색상을 계산한다.
- 감쇄에 따른 최종 색상은 아래와 같이 계산한다. 분모의 0.01의 경우 수치적 안정성 확보의 목적으로, 0으로 나누는 경우를 방지하기 위해 삽입한다.
 - ◆ 아래 수식에서 attenuationFactor는 PointLight의 속성으로 uniform으로 입력 받으며, d는 point light와 fragment 사이의 거리이다.

$$\text{Attenuated Color} = \frac{\text{color}}{(\text{attenuationFactor} * d^2 + 0.01)}$$

where d is a distance between fragment and point light source

- Point Light 클래스는 Directional Light와 마찬가지로 Light 기본 클래스를 상속받아 유도 클래스로 구현하였다. PointLight.js를 참고하시오.
- 구현이 올바르게 이루어졌는지 확인하기 위해 main.js에서 정의한 세 개의 Point light 중, 빨간색 광원에 대해 x,y,z위치와 attenuationFactor를 조정할 수 있는 슬라이더와 콜백 함수를 구현해 두었다. 올바르게 구현된 경우, 첨부한 결과 영상과 같이 빨간색 광원의 변화에 따라 장면이 다르게 그려진다.