

# 컴퓨터 그래픽스

## OpenGL 은면 제거

2016년 2학기

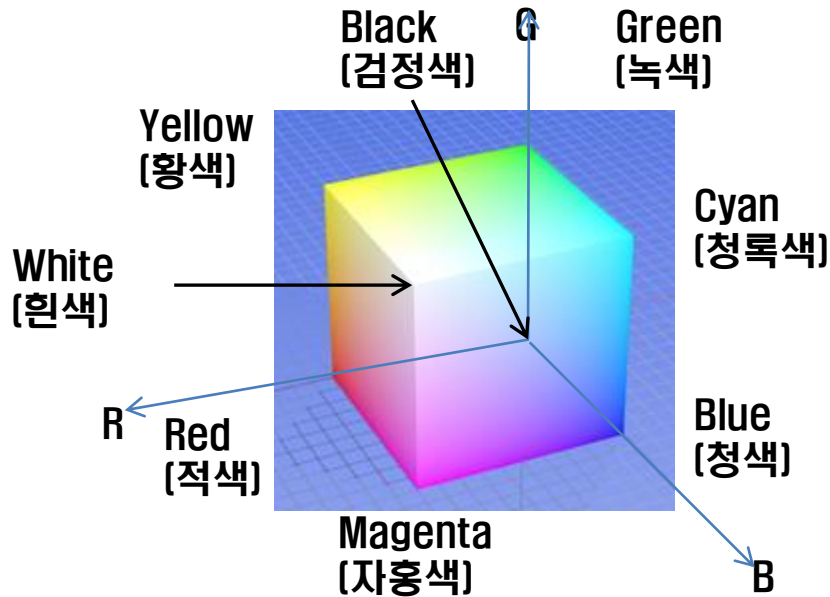
## 2. OpenGL 은면 제거 내용

---

- 은면 제거
  - 컬러
  - 셰이딩
  - 상태 설정
  - 은면 제거

# 컬러와 조명 효과

- openGL에서의 컬러 모델: RGB color model



- 그림색 결정
  - glColor<x><t> (red, green, blue, alpha);
    - x: 인자의 개수, t: 인자의 자료형
    - 현재 드로잉 색상을 설정, 이 명령 뒤에 드로잉 되는 모든 물체는 이 색상을 사용한다.
      - glColor3f, glColor3ub...
    - glColor 함수는 이 명령 뒤에 그려지는 모든 버텍스에 적용되는 색상을 지정하는데 사용된다.
  - 각 버텍스에 다른 색상을 지정하면 색상이 부드럽게 변환된다.

# 쉐이딩

---

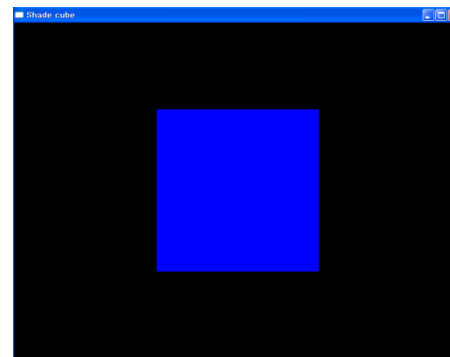
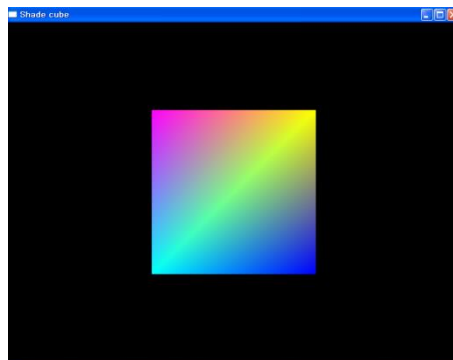
- 음영 (Shading) 넣기

- 쉐이딩: 음영 또는 표면 렌더링으로 물체 면의 색을 부여하는 것
- 쉐이딩 모델 설정
  - `glShadeModel (GLenum mode);`
    - 기본 쉐이딩, 폴리곤 내부의 색은 각 정점에 설정된 색을 보간한 값으로 지정된다. Flat 이나 smooth 으로 결정한다.
    - GLenum mode:
      - » `GL_FLAT`: 평면 쉐이딩 (flat shading) – 마지막 정점에 대해 설정된 색으로 칠해지게 된다.
      - » `GL_SMOOTH`: 부드러운 쉐이딩 (smooth shading) – 지정된 색들이 보간법에 의해 점차로 변해가게 칠해진다.

# 쉐이딩

- 예) 부드러운 쉐이딩이 적용된 사각형 그리기

```
glShadeModel (GL_SMOOTH);           // 또는 glShadeModel (GL_FLAT)
glBegin (GL_QUADS);
    glColor3f (1.0f, 1.0f, 0.0f);    // Yellow
    glVertex3f (100.0, 100.0, 0.0);
    glColor3f (1.0f, 0.0, 1.0f);    // Magenta
    glVertex3f (-100.0, 100.0, 0.0);
    glColor3f (0.0f, 1.0f, 1.0f);    // Cyan
    glVertex3f (-100.0, -100.0, 0.0);
    glColor3f (0.0f, 0.0f, 1.0f);    // Blue
    glVertex3f (100.0, -100.0, 0.0);
glEnd ();
```



# 상태 설정

---

- OpenGL에서의 상태 및 상태 관리
  - 대부분의 상태들은 (라이팅, 텍스처링, 은면 제거, 안개 효과 등) 디폴트로 비활성화(disable)되어 있다.
  - 상태를 활성화(켜거나)하거나 비활성화(끄는)하는 명령어
    - Void **glEnable** (GLenum cap);
      - 지정한 기능을 활성화한다.
    - Void **glDisable** (GLenum cap);
      - 지정한 기능을 비활성화 한다.
  - 활성화 여부를 체크하는 명령어
    - GLboolean glIsEnabled (GLenum cap);

# 상태 설정

---

- `glEnable(GLenum cap)/glDisable(GLenum cap)`
  - OpenGL이 제공하는 다양한 기능들을 켜거나 끄거나 하는 함수
  - Cap:
    - `GL_ALPHA_TEST` : 알파값을 테스트 (`glAlphaFunc`)
    - `GL_BLEND`: 픽셀 블렌딩 연산을 수행 (`glBlendFunc`)
    - `GL_CULL_FACE`: 앞면 혹은 뒷면을 향하는 폴리곤을 선별 (`glCullFace`)
    - `GL_DEPTH_TEST`: 깊이를 비교
    - `GL_DITHER`: 컬러의 디더링 수행
    - `GL_LIGHTx`: 조명x를 켜다
    - `GL_LIGHTING` : 조명 연산을 켜다 (`glLight`)
    - `GL_LINE_SMOOTH`: 선의 안티알리아싱 효과
    - `GL_STENCIL_TEST`: 스텐실 테스트
    - `GL_TEXTURE_CUBE_MAP`: 큐브맵 텍스처링
    - `GL_TEXTURE_1D`: 2D 텍스처링이 생성되지 않을 때 1D 텍스처 맵핑 수행 (`glTexImage1D`)
    - `GL_TEXTURE_2D`: 2D 텍스처 맵핑 수행 (`glTexImage2D`)
    - ...



# 은면 제거

---

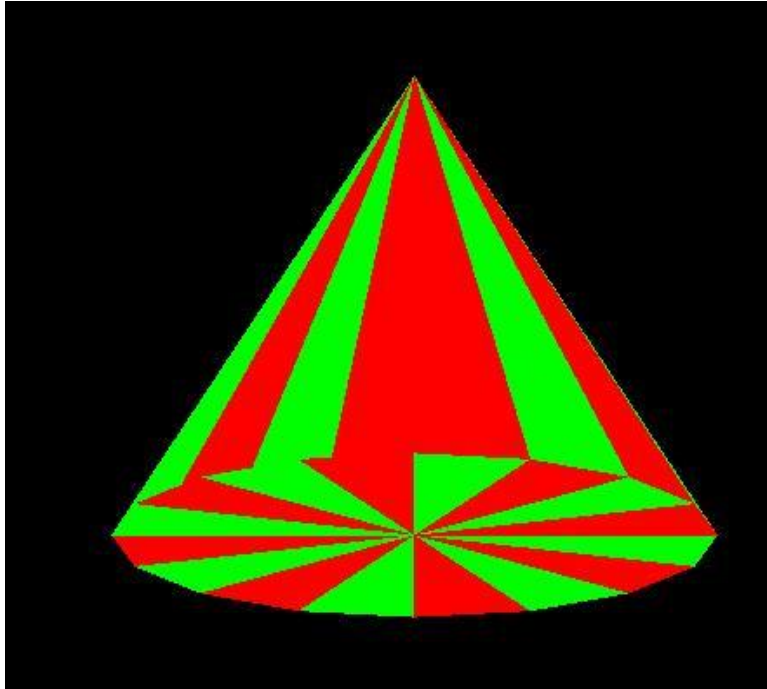
- 은면 제거

- 3차원 장면을 2차원 평면에 투영시키면 물체들이 중첩될 수 있는데, 관측자의 시점에서 가까운 면은 보이고 깊이가 큰 면은 가려져 보이도록 은면 제거를 한다.

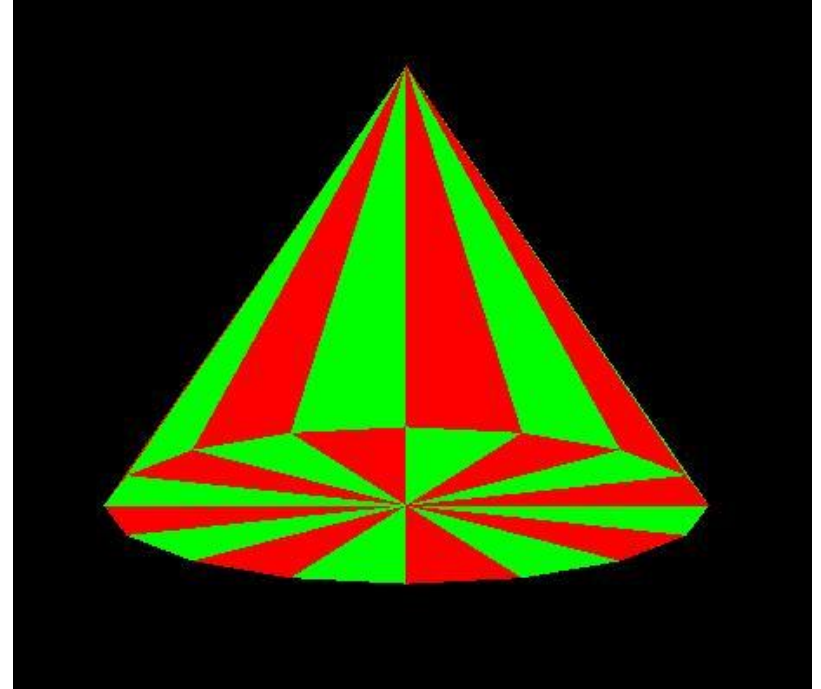
- 깊이 검사 (depth test)를 사용한다.

- 윈도우 초기화 시 깊이 검사 모드 설정
      - `glutInitDisplayMode ( GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH )`
    - 깊이 버퍼를 클리어한다
      - `glClear ( GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT );`
    - 깊이 검사를 설정:
      - `glEnable ( GL_DEPTH_TEST );`
    - 깊이 검사를 해제:
      - `glDisable ( GL_DEPTH_TEST );`

## 은면 제거



Depth test를 안 했을 때



Depth test를 했을 때

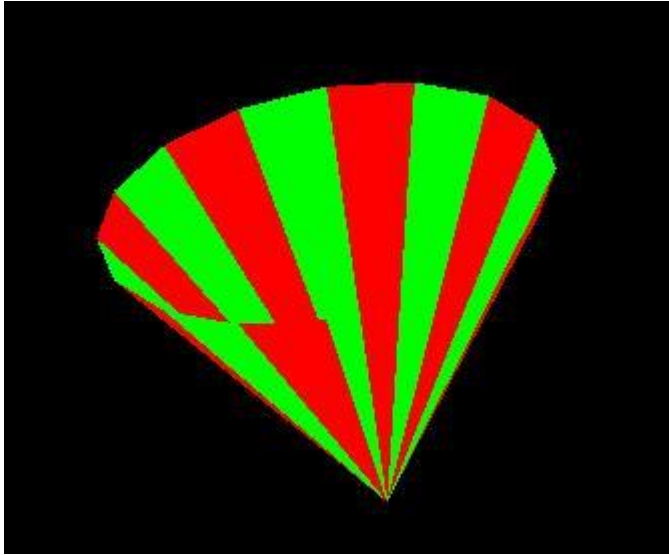
# 은면 제거

---

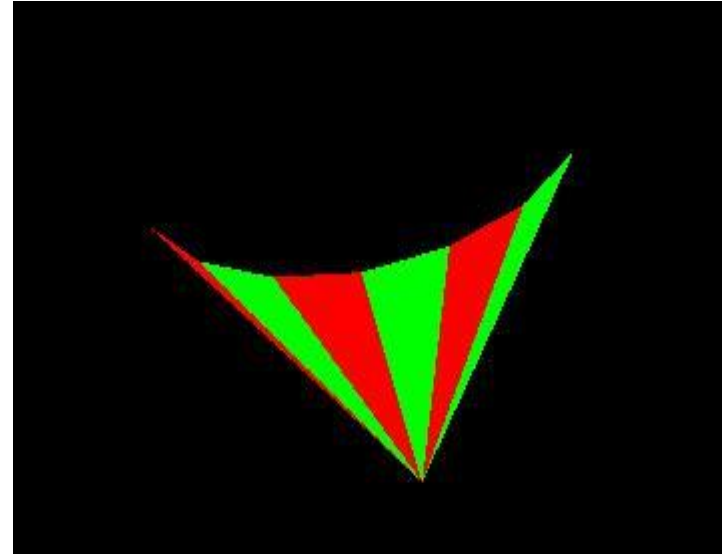
- 컬링 (culling)
  - 후면을 선별(backface culling)하여 뒷면을 모두 제거할 수 있다.
    - Winding을 이용하여 폴리곤의 앞면과 뒷면을 구분한다.
    - 시계 반대방향으로 winding되는 폴리곤이 앞면이다.
  - 컬링 설정: glEnable (GL\_CULL\_FACE);
  - 컬링 해제: glDisable (GL\_CULL\_FACE);
  - void **glFrontFace** (GLenum mode);
    - » 폴리곤의 어느 면이 앞면 또는 뒷면인지 정의한다.
    - » 장면이 달한 객체로 구성되어 있을 때 그 객체의 내부 연산은 불필요한데, 폴리곤의 어느 면이 앞면인지를 결정할 수 있다.
    - » GLenum mode: GL\_CW – 시계방향, GL\_CCW – 반시계 방향
    - » glFrontFace (GL\_CW): 시계 방향을 앞면으로
    - » glFrontFace (GL\_CCW): 반시계 방향을 앞면으로

## 은면 제거

---



Culling을 안 했을 때

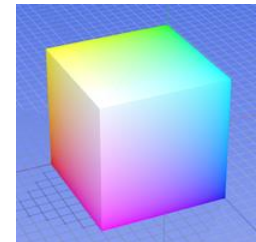


Culling을 했을 때

# 실습 21

- RGB 컬러 모델 만들기

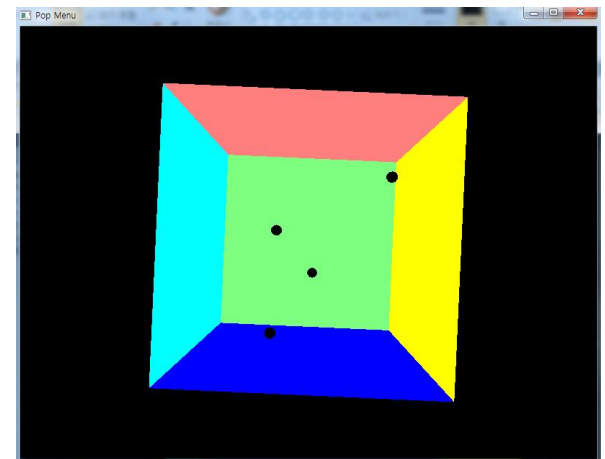
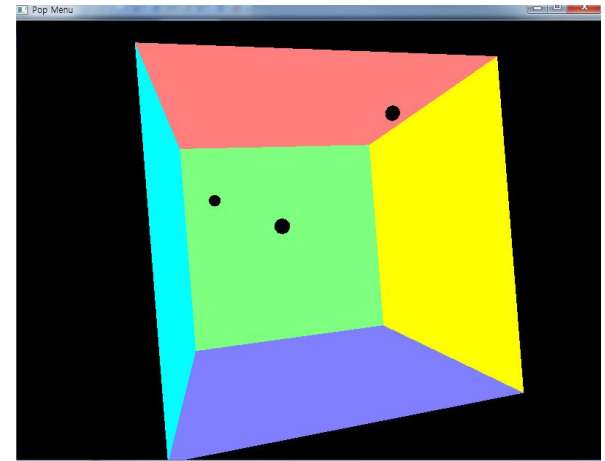
- 육면체를 만들어서 각 꼭지점에 Red, Green, Blue, Yellow, Cyan, Magenta, White, Black의 값을 대응하여 컬러 모델을 만들기
- 육면체는 GL\_QUAD를 사용하여 그리기
- 초기에 X축으로 30도, y축으로 45도로 회전 돼있고, y축에 대하여 회전하고 있도록 한다.
- 팝업 메뉴를 추가하여 은면 제거와 컬링, 쉐이딩을 추가한다.
  - 은면 제거: on / off
  - 컬링: on / off
  - 쉐이딩: flat / smooth
  - 윗면: yes / no (윗면을 그린다 / 윗면을 안 그린다)
    - 윗면에 열리면 열린 면이 수직으로 서 있다.
    - 열리는 모션 구현
  - 앞면: yes / no (앞면을 그린다 / 앞면을 안 그린다)
    - 앞면에 열리면 열린 면이 수평으로 누워있다.
    - 열리는 모션 구현



# 실습 22

## • 3차원 공 튕기기

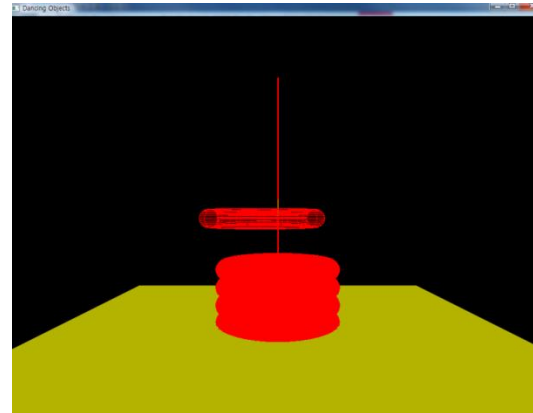
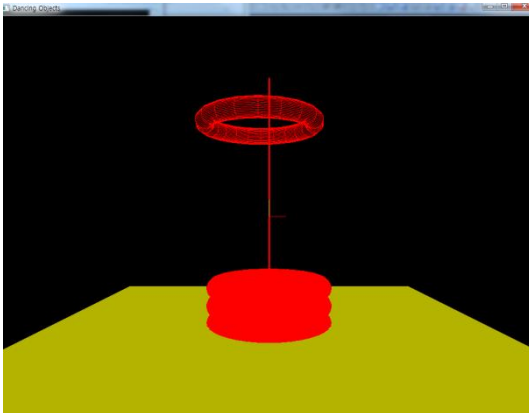
- 원근 투영/은면제거를 사용한다.
- 공을 튕길 공간을 평면을 이용하여 직육면체의 형태로 만든다.
  - 각 면은 다른 색으로 표현한다.
  - 왼쪽, 오른쪽, 아래쪽, 위쪽, 뒤쪽의 면을 그린다.
  - 실습 19는 육면체의 바깥쪽이, 실습 20에서는 안쪽이 보이는 면이다.
- 공간 내부에서 공(sphere)이 이동한다.
- 직육면체의 면에 맞으면 맞은 면은 색을 바꾸고, 공은 방향을 바꿔서 이동을 한다.
- 키보드 명령
  - z/Z: z축으로 양/음 방향으로 이동
  - y/Y: Y축에 시계/반시계 방향으로 회전
  - B: 볼이 새로 생겨서 튕기기 시작한다 (최대 5개)
- 마우스 명령
  - 마우스를 왼쪽으로 옮기면 육면체가 z축 따라 왼쪽으로 회전
  - 마우스를 오른쪽으로 옮기면 육면체가 z축 따라 오른쪽으로 회전



## 실습 23

- 회전하며 내려오는 토러스 그리기

- 막대가 가운데 있고, 토러스가 막대 주위를 빙글빙글 돌면서 내려온다.
- 토러스가 바닥에 닿으면 위에서 토러스가 또 떨어지면 바닥에 있는 토러스 위에 놓인다.
  - 토러스 높이 glutWireTorus 함수의 첫번째 인자가 높이의 절반이다.
  - 막대의 끝까지 닿으면 다시 처음부터 쌓이기 시작한다.
- 2개의 토러스가 내려온 후, 큰 평면 한 개가 회전하면서 내려오고 다시 토러스가 내려온다.
- 명령어에 의해 은면제거 실행/해제를 한다.



## 실습 24

---