

기말고사

담당교수: 단국대학교 멀티미디어공학전공 박경신

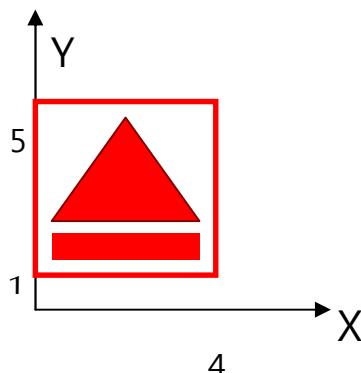
- 답은 반드시 답안지에 기술할 것. 공간이 부족할 경우 반드시 답안지 몇 쪽의 뒤에 있다고 명기한 후 기술할 것. 그 외의 경우의 답안지 뒤쪽이나 연습지에 기술한 내용은 답안으로 인정 안 함. 답에는 반드시 네모를 쳐서 확실히 표시할 것.
 - 답안지에 학과, 학번, 이름 외에 본인의 암호를 기입하면 성적공고시 학번 대신 암호를 사용할 것임.
- 다음은 oglclass에서 제공하는 `matrix4x4` 클래스를 이용하여 변환을 수행한 예를 보여주고 있다. 이 코드에서 밑줄 친 `glMultMatrixf()` 부분을 `glTranslatef()`, `glRotatef()`, `glScalef()` 함수로 대치하여 작성하라 (10점).

```
void display()
{
    glClear(GL_COLOR_BUFFER_BIT);
    glMatrixMode(GL_MODELVIEW);
    drawObject();           // 아래 왼쪽 기본 도형

    matrix4x4 M, T, R, S;
    T.translate(vector3(1, 0, 0));
    R.rotate(90, 'z');
    S.scale(vector3(2, 2, 1));
    M = T * R * S;
    glPushMatrix();
    glMultMatrixf(M);
    drawObject();
    glPopMatrix();

    glPushMatrix();
    glMultMatrixf(M);
    glMultMatrixf(T);
    drawObject();
    glPopMatrix();
}
```

그리고 다음 왼쪽 기본 도형에, 위의 최종 실행 결과를 아래 네모 칸 안에 정확한 척도를 그려서 나타내라 (10점).



2. 다음은 관측 (Viewing) 변환과 투영 (Projection) 변환에 관한 질문이다 (20점).

```
void display()
{
    glClear(GL_COLOR_BUFFER_BIT);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluPerspective(90, 1, 1, 100);

    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
    glTranslatef(0, 0, -10);
    glRotatef(-90, 0, 1, 0);

    glPushMatrix();
    glTranslatef(0, 0, 5);
    glColor3f(1, 0, 0); // 빨간공
    glutWireSphere(1, 16, 16);
    glPopMatrix();
    glPushMatrix();
    glTranslatef(-5, 0, -5);
    glColor3f(0, 0, 1); // 파란공
    glutWireSphere(1, 16, 16);
    glPopMatrix();
}
```

- 1) 위 코드에서 밑줄 친 이동 변환과 회전 변환 함수를 `gluLookAt()` 함수로 대체하려는데, 같은 그림이 나타나도록 하기 위해서 이 함수의 9개 인자를 정확히 기술하라. 어떤 방식으로 문제를 풀었는지 상세히 기술할 것 (10점).

- 2) 위 코드에서 이탈릭체 `gluPerspective()`를 `glOrtho(-20, 20, -20, 20, -20, 20)` 함수로 대체하면, 최종 실행 결과 화면이 어떻게 바뀌는지 그림으로 (빨간공과 파란공의 위치와 크기) 자세히 설명하라 (5점).

//`gluPerspective(90, 1, 1, 100)` 경우

//`glOrtho(-20, 20, -20, 20, -20, 20)` 경우

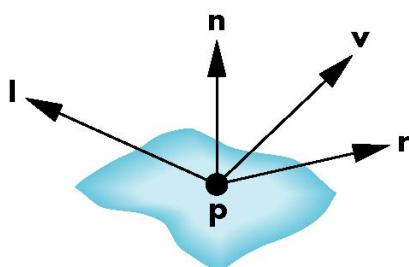
- 3) 그리고 직교 투영 (Orthographic Projection), 축축 투영 (Axonometric Projection), 등축투영 (Isometric Projection), 경사 투영 (Oblique Projection), 원근 투영 (Perspective Projection)의 특징을 자세히 서술하라 (5점).

3. 다음은 조명 (lighting)에 관한 문제이다. 아래의 질문에 답하시오. (20점)

- 1) OpenGL에서 기본으로 사용하는 블린 조명 모델 (Blinn Illumination Model) 공식을 보여주고 있다. 이 공식에서 각 부분이 무엇을 의미하는지 자세히 설명하라 (5점).

$$I = K_a I_a + \sum_{i=0}^{m-1} f_{att}(d) \left\{ K_d I_d (N \bullet L) + K_s I_s (N \bullet H)^n \right\} + E$$

- 2) 블린 모델에서 사용하는 H란 무엇인지 아래 그림에 표시하고 자세히 설명하라 (5점).



3) OpenGL에서 기본으로 사용하는 구로우 세이딩(Gouraud Shading)을 자세히 설명하라. (5점)

4) 만약 OpenGL 프로그램에서 조명 (lighting)을 활성화하고, 광원(light source)과 물체의 재질(material)을 지정했는데 물체에 음영이 제대로 동작하지 않는다면 무엇이 문제인가? 조명에 반드시 필요한 3번째 요소가 무엇인지 자세히 설명하라 (5점).

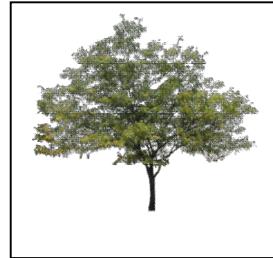
4. 다음은 OpenGL 텍스처 매핑(texture mapping)과 블렌딩 (blending) 프로그램의 일부이다. 아래의 질문에 답하시오. (40점)

```
void initTexture(char * filename, GLuint& textureID, GLuint param)
{
    unsigned char *imageBuffer;
    int imageWidth = 0, imageHeight = 0, numComponents = 0;
    imageBuffer = simage_read_image(filename, &imageWidth, &imageHeight, &numComponents);
    if (imageBuffer != NULL) {
        glGenTextures(1, &textureID);
        glBindTexture(GL_TEXTURE_2D, textureID);
        glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, param);
        glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, param);
        glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_NEAREST);
        glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_NEAREST);
        glTexImage2D(GL_TEXTURE_2D, 0, numComponents == 3 ? GL_RGB : GL_RGBA,
                    imageWidth, imageHeight, 0, numComponents == 3 ? GL_RGB : GL_RGBA,
                    GL_UNSIGNED_BYTE, imageBuffer);
        glBindTexture(GL_TEXTURE_2D, 0);
    }
}
void drawTextureQuad(int textureID)
{
    glTexEnvf(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, GL_MODULATE);
    glEnable(GL_TEXTURE_2D);
    glBindTexture(GL_TEXTURE_2D, textureID);
    glBegin(GL_QUADS);
    glNormal3f(0, 0, 1);
    glTexCoord2f(-1, -1);
    glVertex3f(-1, -1, 0);
    glTexCoord2f(1, -1);
    glVertex3f(1, -1, 0);
    glTexCoord2f(1, 1);
    glVertex3f(1, 1, 0);
    glTexCoord2f(-1, 1);
    glVertex3f(-1, 1, 0);
}
```

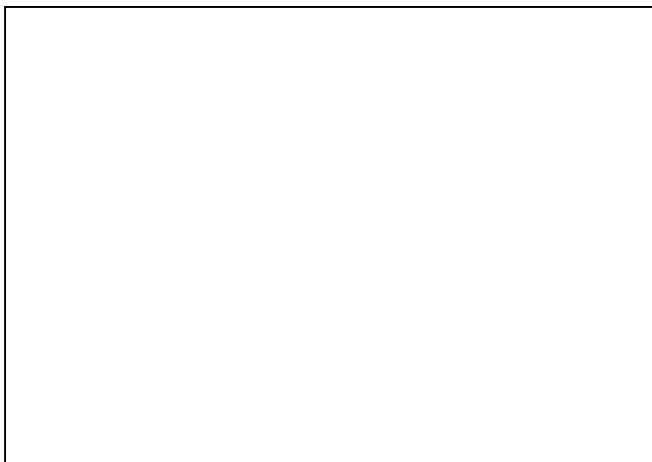
```
glEnd();
glDisable(GL_TEXTURE_2D);
}

void display()
{
    // 중간생략...
    drawTextureQuad(texID);
}

int main(int argc, char *argv[])
{
    // 중간생략...
    initTexture("tree02S.png", texID, GL_CLAMP); // 주변이 투명한 나무 그림
}
```



- 1) 위 코드의 실행 결과를 그림으로 나타내라 (10점).

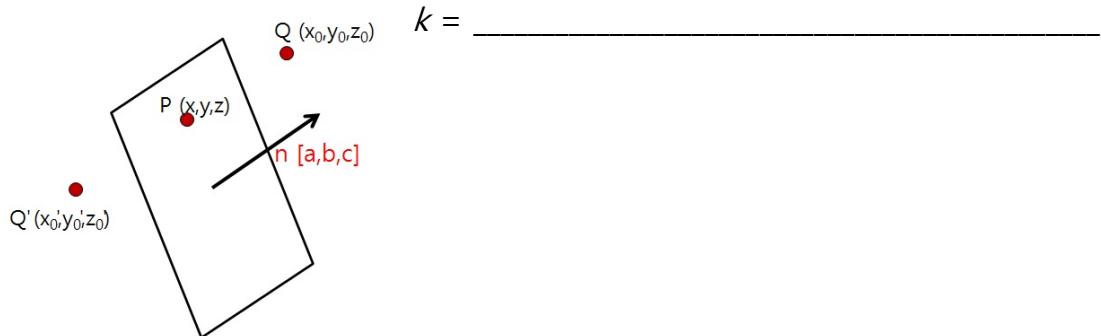


- 2) 만약 flipbook animation 효과를 제공하고 싶다면, 위 코드 중에서 어느 부분 값이 바뀌어야 하는지 설명하라 (5점).
- 3) 만약 텍스쳐 매핑이 거칠게 나타나서 좀 더 부드럽게 나타나게 하고 싶다면, 위 코드 중에서 glTexParameter() 변수를 어떻게 설정해야 할지 자세히 설명하라 (5점).

- 4) 만약 밑맵 (Mipmap) 을 활성화하고 싶다면, 위 코드 중에서 어느 부분이 어떻게 바뀌어야 하는지 자세히 설명하라 (5점).
 - 5) 만약 텍스쳐 맵핑에 조명 (Lighting) 효과를 제공하고 싶지 않다면, 위 코드 중에서 glGetUniformLocation() 환경변수를 어떻게 설정해야 할지 자세히 설명하라 (5점).
 - 6) 위 코드는 투명한 나무 그림으로 나타나지 않는다. 문제가 무엇인지 지적하고 display() 함수의 내부 코드를 수정하라 (10점).

5. 다음 문제에 답하시오. (extra 15점)

평면 $(ax + by + cz + d = 0)$ 의 법선 벡터 $n(a, b, c)$ 가 단위벡터일 때, 평면 밖의 한 점 $Q(x_0, y_0, z_0)$ 에서 평면에 가장 가까운 거리 k 를 구하라.



그리고 점 $Q(x_0, y_0, z_0)$ 가 평면에 반사되는 점 $Q'(x_0', y_0', z_0')$ 공식을 구하라. (힌트 $q' = q - 2kn$)

$$x_0' = \underline{\hspace{100pt}}$$

$$y_0' = \underline{\hspace{100pt}}$$

$$z_0' = \underline{\hspace{100pt}}$$

점 $Q(x_0, y_0, z_0)$ 가 xz 평면에 반사되는 점 $Q'(x_0', y_0', z_0')$ 값을 계산하라.

$$x_0' = \underline{\hspace{100pt}}$$

$$y_0' = \underline{\hspace{100pt}}$$

$$z_0' = \underline{\hspace{100pt}}$$