

그래픽스 강의노트 05 - 프리미티브

강영민

동명대학교

2021년 2학기

- OpenGL은 특정한 하드웨어나 운영체제에 의존하지 않고 다양한 시스템에 이식(移植)될 수 있는 개방형 라이브러리
- OpenGL을 통한 학습은 실시간 그래픽스에 대한 이해를 돕고, 다양한 시스템에 적용가능한 그래픽스 프로그래밍 기술을 습득하게 함

프리티브(primitives) - 1/3

- 그래픽 하드웨어는 프로그래머(programmer)가 지정한 프리미티브(primitive) 설정에 따라 정점의 리스트를 처리
- 프리미티브는 OpenGL이 제공하는 그리기 기본요소
- 입력 정점들을 어떻게 조합할 것인가를 결정
- 프리미티브를 사용하는 방법은 다음과 같다.

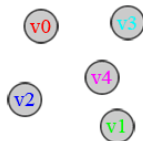
```
glBegin (drawing primitive) ;  
    // vertex position , color , normal , etc  
    glVertexInfo () ;  
glEnd () ;  
}
```

프리티티브(primitives) - 2/3

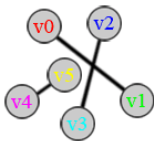
- GL_POINTS: 입력된 정점을 하나씩 점으로 가시화
- GL_LINES: 입력된 정점을 두 개씩 묶어 선분으로 표현
- GL_LINE_STRIP: 입력된 정점을 차례대로 연결하여 하나의 폴리라인 (polyline)을 구성
- GL_LINE_LOOP: 입력된 정점을 차례로 연결한 뒤에 마지막 점을 시작점으로 연결
- GL_TRIANGLES: 입력된 정점을 세 개씩 묶어 삼각형을 그림
- GL_TRIANGLE_STRIP: 처음 세 개 정점으로 삼각형을 그린 뒤, 정점이 추가될 때마다 삼각형을 직전 두 개 정점과 연결하여 삼각형 추가
- GL_TRIANGLE_FAN: 부채 모양으로 삼각형을 추가해 나감
- GL_QUADS: 정점 네 개씩을 묶어 사각형 그리기
- GL_QUAD_STRIP: 처음 네 개 정점으로 사각형 그리고, 이후 두 개씩 묶어 직전 두 개 정점과 함께 사각형 그리기
- GL_POLYGON: 입력된 모든 정점으로 다각형을 그림

프리티브(primitives) - 3/3

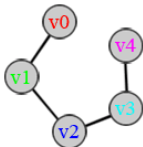
Geometric Primitive Types in OpenTK.OpenGL (defined Clockwise)



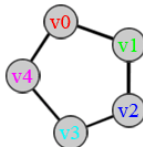
Points



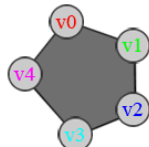
Lines



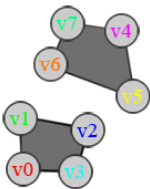
LineStrip



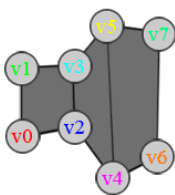
LineLoop



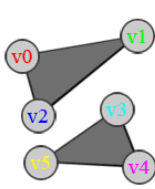
Polygon



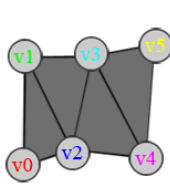
Quads



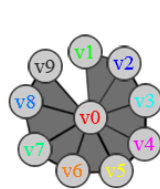
QuadStrip



Triangles



TriangleStrip



TriangleFan

정점 데이터 설정 방법

- 정점 데이터는 위치와 법선벡터, 색 등
- 정점의 위치만을 입력한다면 glVertex[dim-type]으로 입력
- 3 차원 정점의 각 성분을 부동소수점 표현으로 넣는다면, glVertex3f(x,y,z)와 같이 입력
- 다음과 같은 같은 여러 표현이 가능하다.

```
ix : int = -1
iy : int = 0
iz : int = 0
fx : float = 1.0
fy : float = 0.0
fz : float = 0.0
fverts = [fx-1.0, fy+1.0, fz]
```

```
glColor3f(0.5, 0.5, 0.8)
glBegin(GL_TRIANGLES)
glVertex3i(ix, iy, iz)
glVertex3f(fx, fy, fz)
glVertex3fv(fverts)
glEnd()
```

프리미티브 바꾸어 그려보기

Lines 1-25 / 175

```
from OpenGL.GL import *
from OpenGL.GLU import *

import sys

from PyQt5.QtWidgets import QOpenGLWidget, QApplication, QMainWindow,
    QVBoxLayout, QHBoxLayout, QWidget
from PyQt5.QtWidgets import QGroupBox, QComboBox, QPushButton
from PyQt5.QtCore import *
from PyQt5.QtGui import QPainter, QPen

import numpy as np

PRIMITIVES = [ 'GL_POINTS', 'GL_LINES', 'GL_LINE_STRIP', 'GL_LINE_LOOP',
    'GL_TRIANGLES', 'GL_TRIANGLE_STRIP', 'GL_TRIANGLE_FAN',
    'GL_QUADS', 'GL_QUAD_STRIP', 'GL_POLYGON' ]

PRIMITIVE_VALUES = [GL_POINTS, GL_LINES, GL_LINE_STRIP, GL_LINE_LOOP,
    GL_TRIANGLES, GL_TRIANGLE_STRIP, GL_TRIANGLE_FAN,
    GL_QUADS, GL_QUAD_STRIP, GL_POLYGON]

selected = 0

POINTS = [[0, 0], [10, 10], [100, 50]]

class MyGLWidget(QOpenGLWidget):
```

프리미티브 바꾸어 그려보기

Lines 26–50 / 175

```
def __init__(self, parent=None):
    super(MyGLWidget, self).__init__(parent)
    self.colors = []
    self.colors.append(np.array([0.0, 0.0, 0.0]))
    self.colors.append(np.array([0.0, 0.0, 0.0]))
    self.colors.append(np.array([0.0, 0.0, 0.0]))

def initializeGL(self):
    # OpenGL 그리기를 수행하기 전에 각종 상태값을 초기화
    glClearColor(0.8, 0.8, 0.6, 1.0)
    glPointSize(4)
    glLineWidth(2)
    glEnable(GL_BLEND)

def resizeGL(self, width, height):
    # 카메라의 투영 특성을 여기서 설정
    glMatrixMode(GL_PROJECTION)
    glLoadIdentity()
    glOrtho(0, 240, 380, 0, -1, 1)

def paintGL(self):
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT)
    glMatrixMode(GL_MODELVIEW)
    glLoadIdentity()
```


프리미티브 바꾸어 그려보기

Lines 51-75 / 175

```
# 색과 프리미티브를 이용한 객체 그리기
glColor3f(1, 0, 0)
glBegin(GL_POINTS)
for i in range(len(POINTS)):
    glVertex2fv(POINTS[i])
glEnd()

glBegin(PRIMITIVE_VALUES[selected])
nPoints = len(POINTS)
for i in range(nPoints):
    glColor3f((1-i/nPoints), 0, i/nPoints)
    glVertex2fv(POINTS[i])
glEnd()

glColor4f(0, 1, 0, 0.1)
glBegin(GL_LINE_STRIP)
for i in range(len(POINTS)):
    glVertex2fv(POINTS[i])
glEnd()

# 그려진 프레임버퍼를 화면으로 송출
glFlush()
```

프리미티브 바꾸어 그려보기

Lines 76-100 / 175

```
class MyWindow(QMainWindow) :  
  
    def __init__(self, title=''):  
        QMainWindow.__init__(self) # call the init for the parent class  
        self.setWindowTitle(title)  
  
        ### GUI 설정  
  
        central_widget = QWidget()  
        self.setCentralWidget(central_widget)  
  
        gui_layout = QHBoxLayout() # CentralWidget에 사용될 수직 나열 레이아웃  
        # 배치될 것들 - GL Window + Control  
        central_widget.setLayout(gui_layout)  
  
        self.glWidget = MyGLWidget() # OpenGL Widget  
        gui_layout.addWidget(self.glWidget)  
  
        self.controlGroup = QGroupBox('Vertex and Primitives')  
        gui_layout.addWidget(self.controlGroup)  
  
        control_layout = QVBoxLayout()  
        self.controlGroup.setLayout(control_layout)  
        primitive_selection = QComboBox()  
        for i in range(len(PRIMITIVES)):
```

프리미티브 바꾸어 그려보기

Lines 101-125 / 175

```
        primitive_selection.addItem(PRIMITIVES[i])

# ComboBox에 기능 연결
primitive_selection.currentIndexChanged.connect(self.selectPrimitive)

reset_button = QPushButton('reset vertices', self)
reset_button.clicked.connect(self.resetPoints)

control_layout.addWidget(primitive_selection)
control_layout.addWidget(reset_button)

self.canvas = Drawer(parent=self)
gui_layout.addWidget(self.canvas)

def selectPrimitive(self, text):
    global selected
    selected = int(text)
    self.glWidget.update()

def resetPoints(self, btn):
    global POINTS
    POINTS = []
    self.glWidget.update()
    self.canvas.update()
```

프리미티브 바꾸어 그려보기

Lines 126–150 / 175

```
class Drawer(QWidget):
    def __init__(self, parent=None):
        QWidget.__init__(self, parent)
        self.parent = parent
        self.painter = QPainter()

    def paintEvent(self, event):
        global POINTS

        self.painter.begin(self)
        self.painter.setPen(QPen(Qt.red, 6))

        for i in range(len(POINTS)):
            self.painter.drawPoint(POINTS[i][0], POINTS[i][1])

        self.painter.setPen(QPen(Qt.blue, 2))
        for i in range(len(POINTS) - 1):
            self.painter.drawLine(POINTS[i][0], POINTS[i][1], POINTS[i +
1][0], POINTS[i + 1][1])
            self.painter.end()

    def mousePressEvent(self, event):
        POINTS.append([event.x(), event.y()])
        print(event.x(), event.y())
        self.parent.glWidget.update()
```

프리미티브 바꾸어 그려보기

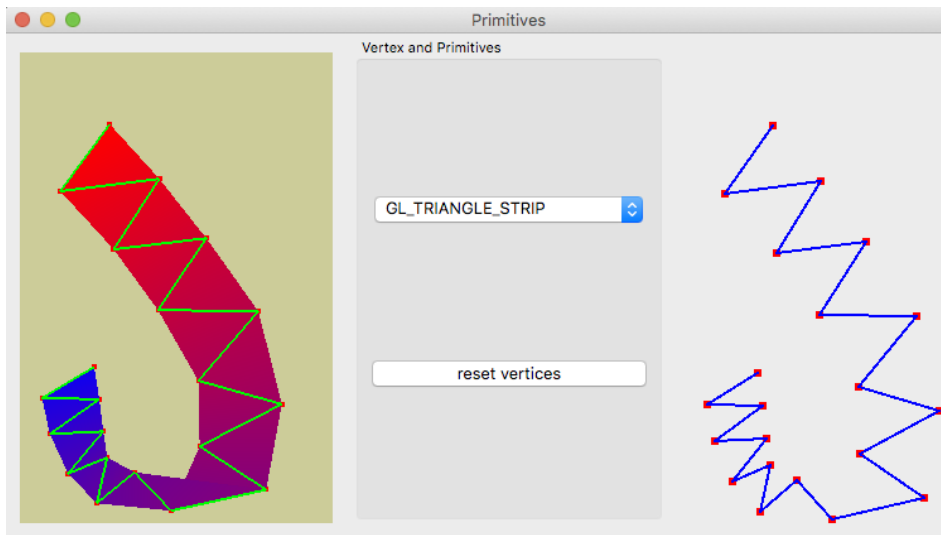
Lines 151-175 / 175

```
self.update()
```

```
def main(argv=[]):  
    app = QApplication(argv)  
    window = MyWindow('Primitives')  
    window.setFixedSize(800, 400)  
    window.show()  
    sys.exit(app.exec_())
```

```
if __name__ == '__main__':  
    main(sys.argv)
```

다양한 프리미티브를 이용한 풍경 그리기



다양한 프리미티브를 이용한 풍경 그리기

다음과 같은 그림을 그려 보자.

