

11. Specular mapping - obecność lub "silla" wzdłużku ^{przebieg} zależy od wartości odrywanej z mapy wzdłużku, czyli odświetlenia tekstury. Musi być obliczone per pixel.

2% 8. Zaznacz własności macierzy ortonormalnych, które są prawdziwe

- ✓ macierz transponowana to macierz odwrotna
- ✗ macierz transponowana to macierz symetryczna do odwrotnej
- ✓ zachowują rozmiar obiektów
- ✓ zachowują rozmiar obiektów w kierunku osi Z
- ✓ równoważne z kwaternionami jednostkowymi

pozostaje: ✗

modelu - ~~zobacz~~
(z w. w. sceny do kamery - nie)

3% 9. Wskaż z jakiego do jakiego układu współrzędnych transformuje macierz

widoku : w. w. sceny → w. w. kamery

modelu : w. w. modelu → w. w. sceny

7 10. Do czego służą funkcje GLSL:

1% dot - iloczyn skalarny dwóch wektorów

1% normalize - normalizacja wektora ($\vec{v} / \text{length}(\vec{v})$)

5% 11. Omów krótko ideę specular mapping.

2% 12. W którym shaderze należy zaimplementować teksturowanie? Zaznacz odpowiedź.

VS FS/PS

5% 13. Na czym polega efekt cross hatchingu. Wyjaśnij w trzech zdaniach.

5% 14. Napisz funkcję GLSL, która zmienia miejscami składowe R i B koloru, a składową przezroczystości ustawia na 50%

vec4 f(vec4 kolor)

```

{
  vec4 _kolor = kolor.bgra;
  _kolor.a = 0.5;
  return _kolor;
}

```

alt. return vec4(kolor.bgr, 0.5);

5	90% - 100%
4	80% - 90%
4	70% - 80%
3+	60% - 70%
3	50% - 60%

Część praktyczna

55%

A. W projekcie pobranego z podanej strony zaimplementuj model oświetlenia Phong. Wzrostki są wyposażone w położenie i normalną.

40%

B. Napisz funkcję, która zmieni kolor na kolor ze skali szarości w negatywie.

5%

C. Napisz funkcję redukującą liczbę kolorów (cel shading, 3 wartości dla każdej składowej).

10%

13. Cross hatching - technika cieniowania, w której ~~stare~~ jednostki ~~zadają~~ wyznaczone jest przez "kreski" na powierzchni obiektów. Linia kreski odpowiada 1-jednostki. Implementowane w pixel shaderze.

Grafika 3D – Nowoczesny OpenGL (2016)

Część teoretyczna

1% ① Napisz krótko do czego służą funkcje OpenGL:

`glGetUniformLocation` – pobranie adresu zm. uni'form w shaderze
`glUniform*` – zmiana wartości zm. uni'form o znanym adresie
`glUseProgram` – wskazanie shadera, który ma być użyty w punkcie renderowania.

2% ② Czy macierz rzutowania w grafice 3D zmniejsza rozmiar transformowanego wektora?

tak nie

Czy macierz rzutowania na płaszczyznę zmniejsza rozmiar transformowanego wektora?

tak nie

2% ③ Ile współrzędnych mają wektory w grafice 3D we współrzędnych jednorodnych?

Napisz wzór redukujący współrzędne jednorodne do „zwykłych” współrzędnych 3D (kartezjańskich).

4 wsp.
(x, y, z, w)

$$\begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} \rightarrow \begin{pmatrix} x/w \\ y/w \\ z/w \end{pmatrix}$$

5% ④ Wymień trzy komponenty oświetlenia w model Phong'a (bez części emisyjnej).

Wskaż komponenty zależące od położenia źródła światła i/lub kamery.

- św. ~~propagowane~~ ^{oświeczone} (ambient) — niezależny od poz. źr. św. i kamery
- św. wyproszone (diffuse) — zależy tylko od poz. źr. św.
- św. w odbłyśku (specular) — zależy ~~tylko~~ od obu.

2% ⑤ Czy złożenie obrotów jest czułe na kolejność operacji? Zaznacz odpowiedź.

tak nie

⑥ Czy współrzędne jednorodne są niezbędne do zapisu poniższych macierzy (zaznacz odpowiedź):

1% a) obrotu tak nie

1% b) translacji tak nie

1% c) rzutowania na płaszczyznę OXY tak nie

1% d) rzutowania perspektywicznego tak nie

5% ⑦ Wpisując cyfry od 1 do 5 ustaw we właściwej kolejności etapy potoku renderowania:

Shader pikseli/fragmentów

4

Shader wertsów

2

Wczytywanie danych z bufora wertsów

1

Test głębi

5

Podział perspektywiczny, przycinanie, rasteryzacja

3