



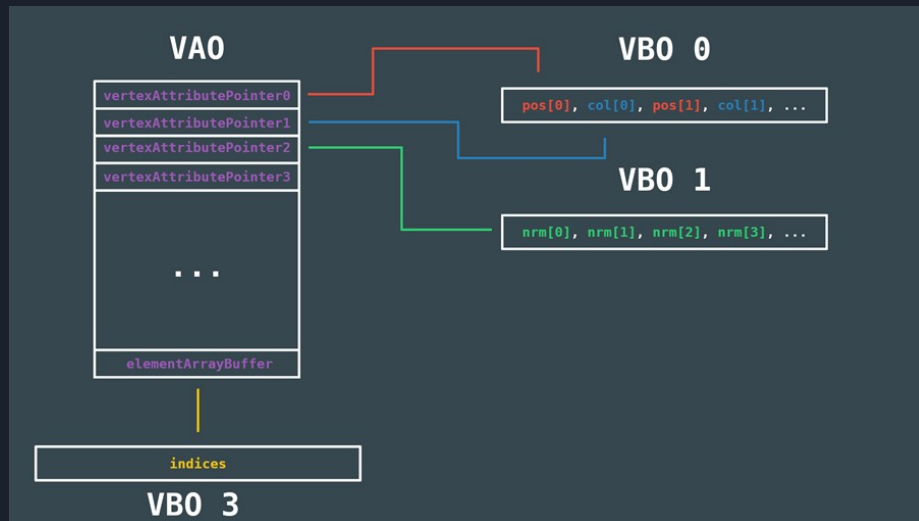
Računarska grafika

Generacija 2023/2024

3. Sedmica

Indeksirano iscrtavanje

S obzirom na to da se ista tjemena ponavljaju u više susjednih poligona, jedan od načina da se smanji broj potrebnih definicija tjemena jeste dodavanje posebnog bafera za indekse (indices) tjemena, i formiranje poligona preko tih indeksa. Ovaj VBO se naziva EBO (Element Buffer Object). Za razliku od ostalih VBO koji idu na Array Buffer, EBO se smješta na Element Array Buffer.





Uniforme

Uniforma je globalna promjenljiva unutar šejdera koju možemo mijenjati direktno iz glavnog programa.

Postavlja se preko *glUniform* funkcija (u zavisnosti od tipa). Prvi parametar ovih funkcija je uvijek lokacija te uniforme u šejderu, a to se saznaje preko *glGetUniformLocation* funkcije. Potrebno je prvo napraviti i uključiti šejder da bi mogli podešavati njegove uniforme.



Providnost

Da bi koristili postigli polu-providnost pri crtanju, potrebno je uključiti Blending. To je poseban korak u rendering pajplajnu koji miješa boju fragmenta koja je trenutno izračunata i na izlazu fragment šejdera sa bojom koja je već sadržana u baferu boja za taj piksel. Ovo se obavlja pozivanjem funkcije *glEnable* sa parametrom *GL_BLEND*, a zatim podešavanjem funkcija miješanja boja sa *glBlendFunc* koja prima dva parametra: koeficijent za boju izvora (izlaz fragment šejdera) i koeficijent za boju odredišta (fragment preko koga crtamo).

Npr. da bi iskoristili alfa kanal izvorne boje u tu svrhu, izvornu funkciju postavljamo za *GL_SRC_ALPHA*, a odredišnu za *GL_ONE_MINUS_SRC_ALPHA*.



Viewport

Viewport je dio prozora na kome crtamo, ograničen od -1 do 1 po osama. Po pravljenu prozora, pravi se jedan viewport koji zauzima čitav inicijalni prozor. Definisanje proizvoljnog viewporta se radi funkcijom *glViewport(x, y, širina, visina)* gdje se x i y odnose na donji lijevi ugao.

glViewport se može iskoristiti za ažuriranje “platna” ukoliko dođe do promjene veličine prozora:

```
void framebuffer_size_callback(GLFWwindow* context, int width, int height)
{
    glViewport(0, 0, width, height);
}
...
glfwSetFramebufferSizeCallback(context, framebuffer_size_callback); //nakon pravljenja konteksta
```



Korisne funkcije/promjenljive

- Veličina jedne tačke nacrtane kao `GL_POINTS` primitiva se može promijeniti sa *`glEnable(GL_PROGRAM_POINT_SIZE);`*
`glPointSize(veličina);`
- Debljina linije se može promijeniti sa *`glLineWidth(veličina);`*
- *`glfwGetTime()`* - vraća vrijeme proteklo od postavljanja tajmera sa *`glfwSetTime()`*, ili pokretanja programa
- *`glPolygonMode(lice, način)`* - Način crtanja poligona (koristan za Wireframe prikaz)
 - Lice: `GL_FRONT`, `GL_BACK`, `GL_FRONT_AND_BACK`
 - Način: `GL_POINTS`, `GL_LINE`, `GL_FILL`
- Fragment šejder:
 - *`gl_FragCoord`* - koordinata fragmenta u rezoluciji prozora (potrebno ručno svesti na opseg `[-1,1]`)
 - *`discard`* - odbacuje fragment (ne crta ništa na tom fragmentu)
 - *`mod(A,B)`* - modulo operator (`A%B`)



Zadatak № 1

Napisati program koji crta ljubičasti, jednakokraki trougao u gornjem desnom kvadrantu prozora, na tamno sivoj pozadini. Baza trougla je duga 0.3 dužine jednog kvadranta prozora, a visina je 0.5 dužine jednog kvadranta. Pritiskom na dugme B, pozadina mijenja boju u bijelu, a pritiskom na dugme S se vraća na tamno sivu. Program se zatvara pritiskom na dugme ESC.



Zadatak № 2

Napisati program koji na lijevoj polovini ekrana crta zastavu države po proizvoljnom izboru, a na desnoj polovini prozora crta zastavu Japana.



Zadatak № 3

Napisati program koji crta šarenu traku proizvoljnih boja okolo ivica prozora koji je dimenzija 500x500, a na sredini ekrana prvo slovo svog imena proizvoljne boje na svijetlo sivoj pozadini. Širina trake iznosi 30% širine jednog kvadranta. Debljina linija slova mora biti veća od 1 piksela. Pozadina se tokom vremena smjenjuje od potpuno bijele do potpuno crne boje.



Zadatak № 4

Nacrtati proizvoljnu plavu zvijezdu na proizvoljnoj poziciji.

Nacrtati poluprovidan kvadrat na centru ekrana, čija je stranica duga 20% jednog kvadranta i implementirati sledeće funkcionalnosti tastera:

ESC - Zatvara program;

Strelice (Arrow keys) - Pomjeranje kvadrata, tako da kvadrat ne može preći ivice prozora;

R - Vraća kvadrat na centar ekrana;

1 - Prikazuje samo tjemena poligona koji čine kvadrat, koja su povećana da budu lakše uočljiva;

2 - Prikazuje ivice poligona (tzv. Wireframe mode);

3 - Prikazuje kvadrat ispunjen bojom;

4 - Isključuje providnost kvadrata

5 - Uključuje providnost kvadrata

Kvadrat se crta preko trougla, i na trougao ne utiču funkcionalnosti tastera 1-5.



Zadatak № 5

Napisati fullscreen aplikaciju koja prikazuje crvenu tačku na centru ekrana (Tačka je veća od 1 piksela) koja može da se pomijera po ekranu pomoću WSAD tastera, a ukoliko izađe izvan ekrana, pojavljuje se na suprotnoj ivici. Boja tačke pulsira od crvene do žute tokom vremena, i pulsira brže što je dalja od koordinatnog početka. Tačka ostavlja trag trenutne boje za sobom. Taster ESC zatvara program. Taster R briše ekran i resetuje tačku na koordinatni početak.



Zadatak № 6

Nacrtati bijeli trougao na crnoj pozadini u gornjem lijevom kvadrantu ekrana, koji se pomijera po ekranu sa WSAD tasterima, a vraća na koordinatni početak tasterom R. Dio trougla koji pređe u gornji desni kvadrant biva obojen crvenom bojom, dio koji pređe u donji desni dio biva obojen ljubičastom bojom, a dio koji pređe u lijevi donji kvadrant biva obojen plavom bojom. Dio trougla koji pređe na donju polovinu ekrana se crta u vertikalnim prugama. Ukoliko trougao priđe donjem desnom uglu, ili se pritisne taster ESC, program se zatvara. Taster R postavlja trougao nazad na početni položaj.