

Ovdje su dane osnovne upute kako biste što lakše izradili prikazanu 3D scenu koja je stavljena na Moodle.

- Najprije morate instalirati **Python** i njegov modul **Pyglet**. Ukoliko imate nekih problema kod instalacije, javite se svojem asistentu za pomoć. Iako Python ima dosta drugih jako korisnih modula od kojih većinu imam instaliranu na svojem računalu, pretpostavljam da oni nisu potrebni u izradi ove zadaće jer nisu uopće korišteni u izradi 3D scene koja je na Moodleu, osim ako Pyglet ne poziva implicitno neki od modula. Primjer je napravljen u Pythonu 2.5.1, ali ne bi trebalo biti problema sa bilo kojom od verzija 2.5.x. Za lakši rad s Pythonom preporučam vam jedan od solidna dva besplatna editora **SciTe** ili **DrPython**. Vjerojatno za **DrPython** trebate instalirati **wxPython** (to je modul za izradu windows aplikacija), a on je korišten kod izrade **DrPython-a**, ukoliko se odlučite za **DrPython**. Nadam se da drugih problema ne bi trebalo biti vezanih uz instalaciju. Ukoliko bude, obratite se za pomoć.
- Svejedno je da li ćete napraviti verziju jedan u kojoj treba neprestano pritiskati tipke da biste se kretali oko objekta ili verziju dva u kojoj je dovoljno samo držati cijelo vrijeme pritisnutu određenu tipku za kontinuirano kretanje. To neće utjecati na bodove. Po mojem subjektivnom mišljenju lakše je napraviti prvu verziju, ali niti druga nije puno kompliciranija, samo treba malo programski proširiti klasu događaja koja prati zbivanja na tastaturi.
- Prije nego što uopće krenete nešto raditi, svakako je neophodno da proučite drugo i treće poglavlje u vodiču kroz **OpenGL** programiranje koji je stavljen na Moodle. U tim poglavljima je objašnjeno kako se grade složeniji objekti pomoću već ugrađenih jednostavnijih objekata u **OpenGL-u**. Isto tako, objašnjeno je na koji način postaviti kameru, odabrati perspektivu i slične stvari koje su neophodne da biste svoju 3D scenu vidjeli na monitoru.
- Prvi važni korak je da izgradite jedan oktaedar pomoću jednostavnijih već implementiranih objekata u **OpenGL-u**. Napominjemo da ne smijete koristiti već možda neku gotovu funkciju koja konstruira oktaedar ukoliko na nju naiđete. Sami morate od samog početka graditi svoj oktaedar ili napraviti svoju funkciju koja će to raditi. Isto tako morate ubaciti i jednu ravninu (tj. omeđeni dio ravnine) na kojoj će oktaedar ležati kako ste vidjeli na priloženom primjeru.
- Proučite peto poglavlje u vodiču kroz **OpenGL** programiranje koje govori o zrakama svjetlosti u 3D sceni. Da bi vaši 3D objekti lijepo izgledali i bili što realniji, potrebno je dobro postaviti svjetla. Vi u svoju scenu morate postaviti dva svjetla, jedno mora biti "usmjereno" (directional light) koje je u beskonačnosti, a drugo neka bude pozicijsko koje je postavljeno na određenu poziciju u prostoru i ima isto određeni smjer i neka bude **spotlight**. Poigrajte se malo s raznim parametrima za svjetla kako biste dobili otprilike onako kako je prikazano u primjeru. Jasno, ne mora se baš sve podudarati, ali je bitno da scena lijepo i što realnije izgleda. **OpenGL** dopušta oko sedam različitih izvora svjetla u jednoj sceni, ali to se razlikuje

od implementacije do implementacije. Da bi vaši objekti u sceni bili ispravno osvijetljeni, potrebno je pravilno postaviti normale u vrhove tih objekata. Kako su svi vaši objekti u sceni izgrađeni od komadića ravnina, potrebno je u njihove vrhove postaviti normale na te ravnine (pazite, te normale moraju imati jediničnu duljinu, a ukoliko nisu, tada od **OpenGL-a** treba zatražiti da ih normira jer u protivnom objekti neće biti ispravno osvijetljeni). Nadalje, za svaki objekt treba postaviti svojstva materijala, npr. kako će objekt upijati svjetlost, kako će ju reflektirati i slično. Pazite, kada se koriste svjetla, tada **OpenGL** ne koristi boje koje ste eventualno pridružili vrhovima objekta, nego koristi normale vrhova i svojstva materijala kako bi odredio boju objekta. Boje vrhova se koriste jedino ako se u sceni ne koristi svjetla, što kod vas nije slučaj jer morate koristiti svjetla. Matematičku pozadinu svega navedenog možete još potražiti u literaturi koja je stavljena na Moodle kako bi vam bilo jasnije o čemu se radi i kako bi što lakše ispravno sve napravili.

- Da bi vaši objekti još ljepše izgledali, stavite na svakog od njih neku teksturu (ravni-ina i oktaedar neka imaju različite teksture). Zgodne teksture možete skinuti s weba i staviti ih na svoje objekte. Pogledajte deveto poglavlje u vodiču kroz **OpenGL** programiranje koje govori o teksturama. Ne trebate detaljno sve gledati, nego samo najvažnije naredbe jer se u tom poglavlju promatraju teksture koje se programski naprave u **OpenGL-u**, a vi ćete ubaciti već gotovu teksturu izvana. Ovdje je važnije vidjeti kako se u Pygletu mogu na jednostavni način učitati teks-ture i staviti ih na definirane objekte. Pazite da dimenzije tekstura budu potencije broja 2 tako da ne upadnete u neke neočekivane probleme. Ukoliko vaša tekstura nije tih dimenzija, napravite resize u nekom programu. Isto tako, pripazite da tek-stura ne bude prevelike veličine (u primjeru su teksture veličina do 200 kilobajta) jer će se u protivnom brzo potrošiti resursi vašeg računala, osim ako nemate neko superračunalo.
- Konačno, trebate omogućiti kretanje u svojoj 3D sceni. Omogućite kretanje oko oktaedra kako je to prikazano u primjeru pomoću tipaka s lijevom i desnom strelicom, te približavanje i udaljšavanje od oktaedra pomoću gornje i donje strelice. Isto tako omogućite rotiranje oktaedra oko koordinatnih osi (pritom se ne krećete oko oktaedra) kako je to napravljeno u primjeru pomoću tipaka q, w, a, s, y, x (ne morate koristiti baš te tipke, slobodno druge odaberete za pojedini događaj). Isto tako omogućite paljenje i gašenje "usmjerenog" (directional) svjetla (u primjeru je to napravljeno pomoću tipke 0) te paljenje i gašenje pozicijskog svjetla (u primjeru je to napravljeno pomoću tipke 1). Pritom pazite da se i svjetla pravilno kreću (ili možda mirno stoje) tako da se dobije lijepi ugođaj.

Vaša 3D scena se ne mora u potpunosti podudarati s onom iz priloženog primjera (što se tiče boja i samog ugođaja), ali je bitno da sve funkcionira na način kako je prikazano u tom primjeru. Ukoliko želite obogatiti svoju scenu još nekim objektima, slobodno to napravite, ali ako i ne napravite, to neće utjecati na bodove. Tri boda za ovu zadaću će se raspodijeliti na sljedeći način:

- pravilno izgrađeni i postavljeni u prostor oktaedar i ravnina na kojoj on leži – 1 bod

- pravilno postavljeno osvjetljenje i teksture na objekte – 1 bod (0.5 boda za pravilno osvjetljenje i 0.5 boda za teksture)
- kretanje u 3D sceni (kretanje oko tetraedra, rotiranje tetraedra, paljenje i gašenje svjetla, zumiranje) – 1 bod

Ukoliko ukupno nećete imati cijeli broj bodova, zaokružiti ćemo na veći cijeli broj. Npr., ako ćete imati 1.5 bodova, upisati ćemo u Moodle 2 boda tako da ne budemo previše strogi s obzirom da se vjerojatno velika većina prvi puta s time susreće.

Nadam se da ćete se zabavljati uz ovu zadaću. Meni je bilo zabavno u izradi onog primjera. Shvatite to kao igru kroz koju primijenjujete teoriju naučenu na predavanjima. Isto tako, slobodno diskutirate na Moodle-u na forumu za diskusiju studenata o eventualnim problemima ili iskustvima vezanim uz ovu zadaću. I mi ćemo biti voljni odgovoriti na vaša pitanja i dati neki savjet, ukoliko ćete imati nekih problema. Nije poanta da tajimo neku naredbu ukoliko ona rješava određeni problem, ali jasno nećemo vam napisati cijeli kod. Želimo da kroz igru i diskusiju svi nešto novo naučimo. Nadamo se da ćete i vi tokom izrade ove zadaće otkriti neke stvari za koje mi ne znamo i podijeliti to s nama. Isto tako, važno je da svatko to sam proba napraviti što bolje može jer tek tada čovjek vidi kako je u teoriji sve divno i lijepo, a u praksi se uvijek mogu javiti neki problemi od kojih neke jednostavno ne možemo izbjeći.

Nakon što na predavanjima odslušate drugo poglavlje o analitičkoj geometriji prostora, tada se odmah možete ozbiljnije prebaciti na rješavanje ove zadaće. Do tada možete malo proučiti `Python` i `Pyglet` i pogledati neke primjere na webu. Zadaću trebate predati najkasnije prije početka petog poglavlja na predavanjima, dakle najkasnije nakon što se obradi u potpunosti četvrto poglavlje (linearni operatori). Još će naknadno biti naveden krajnji datum predaje ove zadaće, što ovisi o tempu kojim će ići predavanja.