

## Računarska grafika

### Domaći zadatak #2 (2012/2013) : 3D grafika - OpenGL

Drugi domaći zadatak je iz oblasti primene grafičke biblioteke OpenGL. Studentima je ponuđeno više zadataka, od kojih treba da odaberu i samostalno izrade jedan. Sve elemente rešenja koji nisu specificirani postavkom, studenti definišu na bazi razumnih, profesionalno opravdanih pretpostavki. **Osim realizacije traženih funkcionalnosti, u ocenu ulazi kvalitet i izgled grafičkog interfejsa.** Postoji mogućnost nadgradnje domaćih zadataka A i B (ne C ili D) sa ciljem da prerastu u završni ili diplomski (master) rad. Nakon uspešne odbrane domaćeg zadatka, studenti zainteresovani za završni ili diplomski rad mogu da se obrate predmetnom asistentu ili nastavniku. Uspešno realizovani završni ili diplomski radovi A ili B bi mogli da imaju i praktičnu vrednost, jer se planira da se kvalitetno realizovan program, na nekomercijalnim osnovama, ponudi institucijama koje rade sa decom i omladinom ometenom u razvoju, u okviru projekta *Lite*.

Za kandidate koji se odluče da rade završne i diplomske radove kao nadgradnju domaćih zadataka A ili B, od interesa je da se usvoji "standardni" (zajednički) format za opis lekcije (XML fajl) kako bi nezavisne aplikacije za pripremu i sprovođenje lekcije koje razvijaju različiti timovi ili pojedinci bile kompatibilne, ali studenti koji se opredele za ovaj zadatak nisu u obavezi da ga koriste (poštuju), odnosno nezavisni timovi i pojedinci mogu koristiti i svoje samostalno razvijene formate lekcije. U prvoj fazi projektovanja, svi studenti koji su zainteresovani za zadatke A ili B radiće na specifikaciji standardnih formata zajednički ili samostalno. Prva faza će biti završena "okruglim stolom" na kojem će učestvovati svi zainteresovani timovi i pojedinci sa svojim predlozima i kritikama drugih predloga. Predmetni asistent će, na bazi zaključaka okruglog stola, odobriti konačnu specifikaciju standardnog formata fajla lekcije, nakon čega timovi i samostalni programeri nezavisno nastavljaju realizaciju zadataka.

### Zadatak A

#### *Gradski prevoz: Razvoj obrazovnog softvera za pomoć u obuci korišćenja gradskog prevoza*

Cilj razvoja je obrazovni softver za simulaciju korišćenja usluga gradskog prevoza. Obučavani treba da nauči kako da obavi sve potrebne radnje poput bezbednog ulaska i izlaska iz vozila. Kretanje se ostvaruje kontrolom figure koju ne treba prikazivati. Od softvera realizovanog kao domaći zadatak se ne očekuje (foto)realističnost prikaza.

Od sredstava za prevoz treba ponuditi autobus i tramvaj, koji se biraju na početku programa. Nakon toga prikazuje se predefinisana scena iz gradskog okruženja koja sadrži jednu zatvorenu jednosmernu putanju duž ulica, za kretanje vozila. Putanja je okružena prizorima grada, poput stambenih zgrada, parkova, itd. Zgrade i stabla nije potrebno detaljno modelirati. Na primer, za zgrade je dovoljno da se predstavljaju kvadrima različitih dimenzija ukrašenih realnim slikama zgrada. Na putanji se nalazi nekoliko stanica na kojima se prevozno sredstvo zaustavlja. Obučavani ne može da se kreće van vozila ili stanice.

Scenario simulacije je sledeći: obučavani na stanici čeka da se vozilo zaustavi i otvori vrata, nakon čega ulazi u vozilo i vrata vozila se zatvaraju. Ukoliko ne zada komandu za ulazak u vozilo na vreme, vozilo odlazi i čeka se sledeće. U slučaju da se obučavani zadrži u zoni vrata, treba da dobije upozorenje da u toj zoni nije bezbedno zadržavanje. Nakon ulaska u vozilo, može da se kreće i da odabere mesto za sedenje.

Nije neophodno detaljno modelirati unutrašnjost i spoljašnjost vozila, ali je potrebno da prozori budu providni (tj. da kroz njih može da se vidi okolina) i da postoje sedišta u vozilu na koja obučavani može da sedne. Potrebno je obezbediti da obučavani ne može da prođe kroz zidove vozila i zatvorena vrata.

Studentima je prepušteno da sami odaberu izgled spoljašnjosti i unutrašnjosti vozila. Vozila mogu da budu niskopodna (bez stepenica). Osim pogleda iz "prvog lica" (tj. animirane figure) obezbediti još tri pogleda: pogled na spoljašnjost vozila (kamera se nalazi ukoso i iznad), pogled na unutrašnjost dotičnog vozila sa prednjeg i zadnjeg kraja vozila.

## **ZAVRŠNI RAD**

Za završni rad potrebne su sledeće dorade.

1. Osim prevoznog sredstva, putanjom se kreću druga vozila (na primer automobili).
2. Scena koja sadrži više putanja kretanja vozila gradskog prevoza koje se ukrštaju (isključivo pod pravim uglom) i imaju zajedničke stanice na kojima obučavani može da preseda. Jednom putanjom se kreće samo jedna vrsta vozila gradskog prevoza. Ukrštanja putanja obezbediti semaforima. Svako vozilo treba obeležiti brojem linije na kojoj saobraća.
3. Stanice treba imenovati, a njihova imena prikazati na tabli istaknutoj u okviru stanice. Ispod imena stanice treba ispisati brojeve linija vozila koja se na toj stanici zaustavljaju.
4. Unutrašnjost vozila gradskog prevoza opremiti sa nekoliko displeja koji obaveštavaju putnike o nazivu stanice kojoj se vozilo približava.

## **DIPLOMSKI (MASTER) RAD**

Za diplomski rad potrebne su sledeće dorade.

1. Posebna aplikacija – editor scene i linija gradskog saobraćaja. Potrebno je omogućiti raspoređivanje zgrada, definisanje ulica, stanica i tramvajskih šina (koje ne moraju biti položene duž ulica). Zgrade se biraju iz predefinisane biblioteke objekata. Nije predviđeno kretanje animirane figure van stanice ili vozila pa nije potrebno dodavati trotoar ulicama. Modeli drugih vozila koja se kreću definisanim putanjama se biraju na slučajan način iz predefinisane biblioteke objekata, a korisnik bira željenu gustinu saobraćaja tako što za svaku putanju zadaje broj vozila koja se njom kreću. Svi elementi se pamte u XML fajlovima lekcija.
2. Režim testiranja snalaženja u gradskom saobraćaju: obučavani dobija zadatak da stigne do određene stanice, polazeći od slučajno izabrane stanice. Obučavani sa sobom nosi plan grada kome u svakom trenutku može da pristupi. Obučavani najpre treba da se korišćenjem plana grada informiše gde se nalazi i gde treba da stigne. Obučavani tada treba da označi stanicu gde se trenutno nalazi i stanicu na koju treba da stigne. Tek nakon uspešne identifikacije polazne i ciljane stanice, obučavani se upućuje na čekanje prevoza. Boduju se radnje vezane za korišćenje prevoznog sredstva: bezbedan ulazak u vozilo, ponašanje u vozilu i bezbedan izlazak iz vozila. Posebno se boduje korektan izbor saobraćajnih linija, presedanje na odgovarajućim stanicama i stizanje na određenu stanicu. Bodovi se upisuju u fajl (formata koji se može jednostavno uvesti u *MS Excel*) i/ili u bazu podataka (eventualno, preko veb-servisa). Bodovanje može biti prikazano ispitaniku, a ne mora. Posebna mala aplikacija ili deo aplikacije Editor za prikaz i analizu bodova.
3. Podržati i postojanje putnika na stanicama i u vozilima. Obučavani ne može da interaguje sa ostalim putnicima.

Moguće su još neke dorade.

## Zadatak B

### *Raskrsnica: Razvoj obrazovnog softvera za obuku snalaženja u saobraćaju*

Cilj razvoja je izrada obrazovnog programa za obuku snalaženja u saobraćaju. Sa jedne strane, program treba da se koristi za obuku dece-pešaka, a sa druge strane da pomaže budućim vozačima motornih vozila da savladaju saobraćajna pravila koja se primenjuju na raskrsnicama koje kontolišu semafori. Od softvera realizovanog kao domaći zadatak se ne očekuje (foto)realističnost prikaza, niti da ispuni postavljeni cilj, već samo da predstavlja osnovu za ciljnu aplikaciju.

Opis izgleda raskrsnice je sledeći:

- Ulice koje čine raskrsnicu se seku pod pravim uglom, dvosmerne su, sa jednom trakom u svakom pravcu
- Na ulicama postoji obeležena horizontalna signalizacija (trake za kretanje vozila, pešački prelaz, linija zaustavljanja vozila ispred semafora, itd.), vertikalna signalizacija (saobraćajni znaci kao što su znak STOP, znak za prvenstvo prolaza, znak za ograničenje brzine, semafor, itd.)
- Deo ulice namenjen kretanju vozila je asfaltiran i odvojen od trotoara posebno obeleženim ivičnjakom. Trotoar može da bude popločan ili presvučen asfaltom koji je svetliji od asfalta na delu ulice namenjenom kretanju vozila. Deo trotoara uz ivičnjak može da bude zelena (travnata) površina
- Ulice treba prikazati na dužini dovoljnoj da se vidi kolona od 10-ak vozila počevši od raskrsnice, u svim pravcima

Ulicama se kreću motorna vozila saglasno svetlosnim signalima sa semafora. Vozila se kreću pravolinijski (ne skreću u poprečnu ulicu na raskrsnici). Program treba da podržava sledeća vozila:

- putnički automobil (3 modela koji se razlikuju po izgledu)
- autobus (1 model)
- kamion (1 model)
- kamion sa prikolicom (1 model)

Kod automobila i kamiona varirati boju karoserije. Vozila ne moraju da se kreću istom brzinom i nemaju isto ubrzanje (kamion sa prikolicom ubrzava sporije od putničkog automobila, na primer). Vozila ne smeju da se sudaraju, a onim vozilima koja usporavaju treba da se upale odgovarajuća crvena svetla. Vozila se povremeno pojavljuju u odgovarajućoj traci na kraju svake od ulica i kreću se u pravcu raskrsnice. Po prolasku kroz raskrsnicu, vozila produžavaju do kraja ulice, a zatim nestaju. U svakom trenutku se ulicama mogu kretati vozila proizvoljnih vrsta.

Svi događaji (pojavljivanje i kretanje vozila, promena svetlosnih signala) se dešavaju bez intervencije korisnika. Korisniku treba omogućiti sledeće poglede za razgledanje trenutne situacije na raskrsnici:

- Pogled odozgo. Kod ovog pogleda, posmatrač se inicijalno nalazi iznad centra raskrsnice, na visini sa koje može da vidi krajeve ulica (maksimalna visina). Posmatrač može da se kreće u vertikalnoj ravni (može da menja visinu).
- Pogled iz sredine raskrsnice. Kod ovog pogleda, posmatrač se nalazi iznad centra raskrsnice, malo iznad visine najvišeg vozila. Posmatrač ne može da se kreće, ali može da menja pravac posmatranja u horizontalnoj ravni i nagib posmatranja u odnosu na horizontalnu ravan (onemogućiti nagibe kojim bi se pogled usmerio ka nebu) .
- Pogled koji prati vozilo. Kod ovog pogleda, posmatrač se stalno nalazi iznad i iza izabranog vozila (pozicija posmatrača prati kretanje vozila). Inicijalno, pogled treba da bude usmeren tako da se vozilo nalazi u vidnom polju posmatrača. Korisniku omogućiti promenu nagiba pogleda. Takođe, korisniku treba omogućiti da bira vozilo nad kojim će se primenjivati ovaj pogled.

Studenti sami biraju početnu i maksimalnu brzinu kretanja svakog vozila, ubrzanje pri kretanju, kao i dužinu trajanja svetlosnih signala (zeleni i crveni signali nisu istog trajanja u obe ulice), tako da program bude interesantan za posmatranje.

Voditi računa da brzina kojom se animira kretanje vozila ili promena svetla na semaforima ne sme da zavisi od brzine računara na kojem se izvršava program.

## **ZAVRŠNI RAD**

Za završni rad potrebne su sledeće dorade:

1. Stvaranje noćnih uslova uvođenjem uličnog osvetljenja i uključivanjem farova na vozilima
2. Prikazivanje zgrada u okolini raskrsnice i duž ulica
3. Kretanje kroz raskrsnicu kada semafor ne radi
4. Ulice sa više traka
5. Mogućnost skretanja vozila u raskrsnici.

## **DIPLOMSKI (MASTER) RAD**

Za diplomski rad potrebne su sledeće dorade:

1. Posebna aplikacija – editor scene (lekcije). Potrebno je omogućiti raspoređivanje zgrada, definisanje ulica i više raskrsnica sa većim brojem pravaca. Ulice koje se završavaju na ivici scene predstavljaju tačke nastajanja i nestajanja vozila. Zgrade se biraju iz predefinisane biblioteke objekata. Modeli vozila se kreću duž ulica, poštujući signalizaciju i ostale učesnike u saobraćaju i mogu da skreću u raskrsnicama (kada se na vozilu uključuje svetlosna signalizacija promene pravca). Moguće je zadati željenu gustinu saobraćaja za svaku tačku izvora saobraćaja (kao broj vozila u minuti). Svi elementi se pamte u XML fajlovima lekcija.
2. Režim testiranja snalaženja pešaka prilikom prelaska raskrsnice: testirano lice upravlja figurom koju nije potrebno prikazivati i dobija zadatak da od zadate početne pozicije na mapi grada stigne do druge pozicije. Obe pozicije se biraju slučajno. U svakom trenutku, osim tokom prelaska ulice, testirano lice može da pogleda plan grada na kome su označene trenutna i ciljna pozicija. U režimu testiranja, dostupan je samo pogled iz prvog lica. Boduju se radnje vezane za bezbedno kretanje: kretanje po trotoaru, prelazak ulice samo na pešačkom prelazu, onda kada je to dozvoljeno saobraćajnim propisima. Bodovi se upisuju u fajl (formata koji se može jednostavno uvesti u *MS Excel*) i/ili u bazu podataka (eventualno, preko veb-servisa). Bodovanje može biti prikazano ispitaniku, a ne mora.
3. Prikazivanje kretanja pešaka po trotoarima i kroz raskrsnice.
4. Podržati i postojanje prioriternih vozila (saobraćajna policija, hitna pomoć, vatrogasci).

Moguće su još neke dorade.

## Zadatak C

### *Odbrana ostrva: razvoj video-igre za jednog igrača*

Cilj razvoja je pravljenje jednostavne video-igre u kojoj se igrač nalazi u ulozi branioca ostrva. Ostrvo napadaju četiri desantna čamca koji ostrvu prilaze konstantnom brzinom iz različitih pravaca. Za odbranu se koristi top, čija kabina može da menja orijentaciju u horizontalnoj ravni (rotacijom oko vertikalne Z ose), a cev može da menja nagib u vertikalnoj (rotacijom oko horizontalne X ose). Zadatak igrača je da granatama ispaljenim iz topa pogodi sva četiri čamca pre nego što oni stignu do ostrva. Broj granata je ograničen. Usvojiti da je jedinica dužine m, a jedinica brzine m/s. Parametre koji ne mogu da se podešavaju od strane korisnika zadati tako da igra bude interesantna (odrediti ih empirijski).

Ovaj zadatak predstavlja unapređenje zadatka radenog na četvrtoj laboratorijskoj vežbi. Pored realizacije svih stavki navedenih u postavci laboratorijske vežbe, potrebno je dodati sledeće funkcionalnosti:

- Preko površi mora prikazati koncentrične krugove koji predstavljaju oznake udaljenosti od ostrva. Poluprečnici krugova su: 50 m, 150 m, 300 m, 500 m, 1000 m.
- Više talasa napada na ostrvo. Novi talas počinje kada igrač potopi sve čamce iz prethodnog talasa. Tom prilikom igrač dobija 10 novih granata. Pravac iz kojeg svaki čamac napada ostrvo slučajno se bira na početku svakog talasa napada.
- Završni meni, koji se prikazuje kada igrač izgubi igru. Meni prikazuje koliko čamaca je igrač potopio i koliko talasa napada je izdržao. Takođe, igraču nudi izbor da se prekine igra ili da se pokrene nova igra.

U slučaju izuzetno kvalitetne realizacije zadatka, prema dogovoru sa predmetnim profesorom ili asistentom, zadatak je moguće proširiti do nivoa diplomskog odnosno master rada.

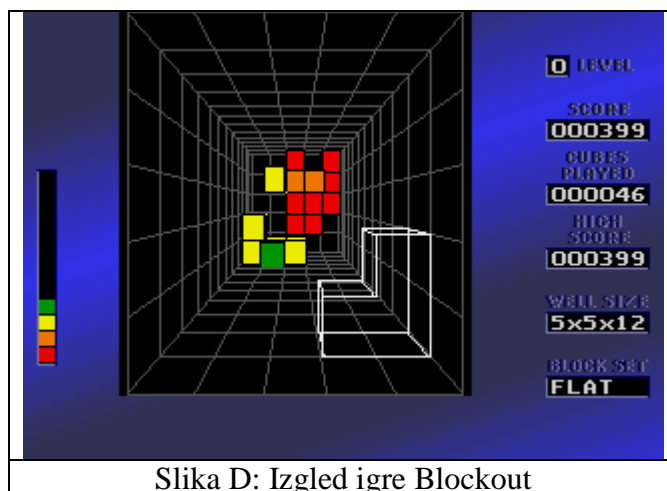
## Zadatak D

### *Blockout*: Razvoj video-igre

*Blockout* je vrlo poznata video-igra, objavljena početkom 90. godina prošlog veka. Predstavlja proširenje legendarne igre *Tetris* u tri dimenzije.

Na slici D (videti ispod) prikazan je izgled ekrana snimljen u toku igre. Od studenata se očekuje da realizuju igru uz korišćenje naprednih mogućnosti biblioteke OpenGL (osvetljenje, providnost, teksture, vrsta materijala, itd) tako da izgled igre bude atraktivan. Obezbediti jedan standardan pogled na "bunar" (kao što je prikazano na slici) i nekoliko pogleda na unutrašnjost bunara sa pokretnom kamerom.

Kod igre *Blockout*, "tabla" za igru se sastoji od "bunara" dubokog  $D$  jedinica dužine, podeljenog na jednake intervale, čije je dno površine  $N \times M$  kvadrata čije su ivice jedinične dužine. Može se smatrati da se zapremina bunara sastoji od  $N \times M \times D$  polja veličine kocke jedinične dužine. Figure su sastavljene od određenog broja kocki stranica jedinične dužine. Tokom igre, figure padaju sa vrha bunara ka njegovom dnu, a zadatak igrača je da ih rasporedi (rotacijom i translacijom) tako da se uklope u figure koje se već nalaze na dnu bunara. Figure ne mogu da se preklapaju, tako da u jednom trenutku jedno polje bunara može da zauzima samo jedna kocka u sastavu jedne figure. Figura koja padne na dno bunara ili na drugu figuru više ne može da se pomera, a nova figura počinje da pada. Dok pada, figura je providna, a kada padne, figura postaje neprovidna. Figura koja sledeća pada se bira na slučajan način. Brzina kojom padaju figure progresivno raste tokom igre. Kada se na nekom nivou dubine bunara popuni svih  $N \times M$  polja, taj nivo bunara se prazni, a sadržaj svih viših nivoa bunara (bližih vrhu bunara) se spušta za jedan nivo niže, prema dnu bunara. Igra je izgubljena kada neka figura viri (u potpunosti ili delimično) van bunara, nakon što se završi njeno propadanje prema dnu bunara. Igrač zadaje vrednosti  $D$ ,  $N$  i  $M$  pre početka igre.



Slika D: Izgled igre Blockout

Za ovaj zadatak nisu predviđene dorade do završnog/diplomskog rada.