

Računarska grafika

Domaći zadatak #2 (2016/2017) : 3D grafika - JavaFX

Drugi domaći zadatak je iz oblasti 3D grafike i rešava se uz primenu grafičkih paketa biblioteke JavaFX. 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 (u izuzetnom slučaju i C ili D) sa ciljem da prerastu u diplomski rad osnovnih studija ili master rad. Nakon uspešne odbrane domaćeg zadatka, studenti zainteresovani za diplomski/master rad mogu da se obrate predmetnom nastavniku. Uspešno realizovani diplomski/master 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 sa posebnim potrebama, u okviru projekta *Lite*.

Za master radove koji se temelje na zadacima A i B, od interesa je da se usvoji "standardni" (zajednički) format za opis lekcije (XML fajl) kako bi nezavisne aplikacije za pripremu (editori) i sprovođenje lekcije (trenažeri), koje razvijaju različiti kandidati, bile kompatibilne. U tom smislu, kandidati se ohrabruju da predlože format XML fajla lekcije. Kandidati koji se opredele za ove zadatke nisu u obavezi da koriste (poštuju) predloženi format, odnosno kandidati mogu koristiti i svoje, nezavisno razvijene, formate lekcije.

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, niti da ispuni postavljeni obrazovni cilj, već samo da predstavlja osnovu za ciljnu aplikaciju.

Od sredstava za prevoz treba ponuditi autobus i tramvaj, koji se biraju na početku programa. Nakon toga se prikazuje 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.

DIPLOMSKI RAD

Za diplomski rad osnovnih studija 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.

MASTER RAD

Za master 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 ciljne 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 odredišnu 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 obrazovni 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.

DIPLOMSKI RAD

Za diplomski rad osnovnih studija 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.

MASTER RAD

Za master 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

Čoveče, ne ljuti se: Razvoj društvene video-igre za više igrača

Napraviti program za igranje društvene igre "Čoveče, ne ljuti se", uz poštovanje svih pravila igre. U igri učestvuje najviše četiri igrača. Pre početka igre, igrači biraju boju i oblik svojih figura. Igrač koji je na potezu najpre "baca" kockicu kliktanjem mišem. Kretanje kockice treba animirati. Nije neophodno poštovati zakone fizike prilikom animacije kockice, već njeno kretanje animirati slučajnom dužinom sekvence slučajnih rotacija i translacija po površini pored table. Kućicu (u kojoj se inicijalno nalaze figure) igrača na potezu treba prikazati svetlijom nijansom boje od one koju je igrač izabrao za boju svojih figura. U zavisnosti od ishoda bacanja kockice, figure igrača na potezu koje je moguće pokrenuti treba da trepere (kontinuirano menjaju nijansu svoje boje) umerenom brzinom. Na isti način treba menjati boju polja na kojima bi se pomenute figure našle ako igrač odluči da ih pokrene. Pomeranje figure koja treperi se vrši klikom miša na nju. Program treba da animira kretanje figure, u vidu niza skokova, odgovarajući broj polja. U slučaju da tom prilikom neka od figura protivničkih igrača biva "pojeden", animirati njen "let" u odgovarajuću kućicu.

Tabla na kojoj se odvija igra je kvadratnog oblika, prekrivena ukrasnom teksturom. U svakom od četiri ćoška se nalazi kućica za jednog od igrača. Kućice igrača koji ne učestvuju treba ostaviti prazne. Dok miruju, figure se nalaze na poljima kružnog oblika.

Potrebno je predvideti najmanje četiri oblika figura koje igrači mogu da biraju na početku igre, s tim što svaki oblik mora da ima kružno postolje, nešto manjeg poluprečnika u odnosu na poluprečnik polja za figure.

Potrebno je omogućiti sledeće poglede (koristiti projekciju sa perspektivom i sprečiti pozicioniranje posmatrača "ispod" table):

- pogled odozgo, kod kojeg posmatrač gleda vertikalno naniže, a može da menja svoju horizontalnu i vertikalnu poziciju
- pogled ka središtu table, kod kojeg se posmatrač kreće po polusferi sa centrom u središtu table, sa mogućnošću promene poluprečnika polusfere. Inicijalno, poluprečnik polusfere treba da bude takav da tabla (sa malim prostorom sa strane na kojem se baca kockica) bude upisana u nju.
- pogled iznad kućice igrača na potezu ka figurama koje su raspoređene po tabli (van kućice). Omogućiti izbor figure koja se fokusira. U slučaju da nema figura u polju, posmatrač gleda u centar table. Pozicija posmatrača treba da bude fiksna i dovoljno uzdignuta iznad ravni table da bi se jasno mogla videti cela tabla.

Kretanje posmatrača i promena pogleda treba da bude nezavisna od kretanja kockice ili figura. Za interakciju prvenstveno koristiti miša, a tastaturu za promenu pogleda ili fokusirane figure.

Za ovaj zadatak nisu predviđene dorade do završnog ili diplomskog rada, izuzev u slučaju izuzetno uspešne realizacije domaćeg zadatka.

Zadatak D

Raketa: Razvoj video-igre za jednog igrača

Zadatak predstavlja proširenje zadatka rađenog na 4. laboratorijskoj vežbi. Navedene stavke predstavljaju proširenja u odnosu na zahteve za 4. laboratorijsku vežbu, čija se implementacija podrazumeva. Parametre koji ne mogu da se podešavaju od strane korisnika zadati tako da igra bude interesantna (odrediti ih empirijski).

Potrebno je realizovati sledeće dodatne funkcionalnosti:

1. Konstruisati jednu novu raketu po sopstvenom izboru i omogućiti igraču da bira između tri rakete pre početka igre (druge dve rakete su već ponuđena raketa u skeletu i raketa iz zahteva lab vežbe). Prilikom biranja, rakete se prikazuju konstantno rotirajući tako da se vide iz svih uglova. Prikazati i parametre rakete: maksimalnu brzinu i brzinu skretanja. Rakete treba da se razlikuju po maksimalnoj brzini i brzini skretanja tako da jedna npr. ima veću maksimalnu brzinu, ali je zato njom teže upravljati, jer joj je brzina skretanja manja. Dodati parametar gorivo kojim raspolaže svaka raketa i koje se troši u zavisnosti od brzine kojom se raketa kreće. Trenutno stanje rezervoara treba prikazivati na ekranu. Ukoliko raketa ostane bez goriva, igra se završava.
2. Dodati deo *HeadUp* ekrana, pored postojećeg "pogleda odozgo", na kome je moguće pratiti vertikalnu poziciju rakete u odnosu na platformu. Na polovini vertikalnog prikaza označiti platformu, a raketu prikazivati kao crvenu tačku na njenoj trenutnoj visini.
3. Na postojećem *HeadUp* ekranu dodati granice prostora obuhvaćenog scenom.
4. Proširiti igru tako da igra ima trajanje. Igrač mehure sakuplja do isteka vremena kada se igra završava i igraču se prikazuje ukupno vreme igranja i broj sakupljenih mehura. Svaki sakupljeni mehur uvećava preostalo vreme za konstantnu vrednost.
5. U prostoru kroz koji se raketa kreće postaviti žetone koji dopunjuju rezervoar sa gorivom za konstantnu vrednost. Prolaskom kroz njih, povećava se nivo goriva u rezervoaru. Crtanje žetona uraditi po sopstvenom izboru.
6. U prostoru kroz koji se raketa kreće postaviti žetone koji trenutno daju veću maksimalnu brzinu raketi. Oni se sakupljaju i aktiviraju pritiskom na neki taster. Njih igrač može koristiti u situacijama kada želi brže da stigne do nekog odredišta. Crtanje žetona uraditi po sopstvenom izboru. Broj sakupljenih neiskorišćenih žetona se prikazuje na ekranu.
7. U prostoru kroz koji se raketa kreće postoje i tela koja treba izbegavati, jer privremeno usporavaju raketu. Crtanje tela uraditi po sopstvenom izboru.
8. Omogućiti sletanje rakete na platformu pritiskom na *Space* ukoliko se raketa nalazi u njenoj blizini. Raketa se spušta u položaj spreman za poletanje. Kada je raketa na platformi, puni joj se rezervoar. Igrač u bilo kom trenutku može ponovo lansirati raketu sa platforme.

Za ovaj zadatak nisu predviđene dorade do završnog ili diplomskog rada, izuzev u slučaju izuzetno uspešne realizacije domaćeg zadatka.