

## **Računarska grafika**

### **Domaći zadatak #1 (2016/2017) : 2D grafika - JavaFX**

Prvi domaći zadatak je iz oblasti 2D 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, B i C, a u slučaju izuzetno kvalitetne izrade i zadataka 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 završni i master rad B bi mogao da ima 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*.

Zadatke A i B, može da radi tim od 2 studenta, tako što jedan radi Zadatak A1/B1 (*Editor*), a drugi Zadatak A2/B2 (*Takmičenje/Trenažer*). Student koji je zainteresovan za kasniju nadogradnju do diplomskog/master rada, treba da radi samostalno Zadatak A1/B1, a kasnije u okviru diplomskog/master rada uradi i Zadatak A2/B2, uz još neke dorade za master rad.

### **Zadatak A**

#### *OlimpijadaZnanja*: Razvoj softvera za sticanje i proveru znanja kroz igru

Cilj razvoja je obrazovna igra za više igrača u kojoj igrači napreduju ka cilju tako što daju tačne odgovore na pitanja. *OlimpijadaZnanja* se sastoji od dve aplikacije: *Editora*, alata kojim se kreira tabla za igru i definišu pitanja i *Takmičenja*, odnosno same igre.

Tabla za igru se sastoji od pozadinske slike preko koje su iscrtana polja po kojima se kreću figure igrača. Igrač koji je na redu da igra baca kockicu, a njegova figura se kreće za odgovarajući broj polja. Polje na koje figura stane može da bude prazno ili da sadrži pitanje koje se postavlja igraču. Ako je polje prazno, sledeći igrač dolazi na red da igra. Ako polje sadrži pitanje, dalji sled događaja zavisi od toga da li igrač odgovori tačno na pitanje, odustane od odgovaranja ili istekne vreme za odgovor i biće objašnjen kasnije. Pobeđuje onaj igrač koji prvi stigne do polja "Cilj".

### **Zadatak A1**

#### *OlimpijadaZnanja - Editor*: Razvoj alata za pripremu takmičenja

Alat za pripremu takmičenja (*Editor*) treba da ima sledeće funkcionalnosti:

1. definisanje skupa pitanja koja će se postavljati igračima
2. unos pozadinske slike table – podržati formate slika (JPG/PNG/GIF/BMP)
3. definisanje polja za igru i mogućih pitanja kojima igrači mogu da se kreću

Prilikom definisanja jednog pitanja, sastavljač unosi sledeće podatke:

- oblast znanja potrebnog da se odgovori na pitanje (geografija, istorija, OO programiranje, računarska grafika, itd); svakoj oblasti se automatski pridružuje boja koju sastavljač može da promeni
- vreme koje igrač ima na raspolaganju da odgovori na pitanje
- težina pitanja (veoma jednostavno, jednostavno, umereno, teško, veoma teško)

- način davanja odgovora na pitanje:
  - unos vrednosti preko tastature
  - izbor jednog (ili više) od ponuđenih odgovora
  - pogađanje odgovora navođenjem slova od kojih se on sastoji: inicijalno su korisniku prikazana samo neka slova odgovora, a ostala su zamenjena praznim poljima, s tim što je jasno naznačeno koliko ima reči, koliko slova ima svaka reč i koja slova su poznata. Zadatak igrača je da, u ograničenom broju pokušaja, pogodi sadržaj odgovora tako što predlaže slova koja nedostaju. Ako predloženo slovo postoji, onda se ono prikaže u odgovarajućim praznim poljima; ako ne postoji, broj pokušaja se smanjuje za jedan. Ako igrač iskoristi sve pokušaje, smatra se da je na pitanje dao netačan odgovor.
  - ishod davanja odgovora ili odustajanja od davanja odgovora

Pitanja i ponuđeni odgovori (ako je to način odgovaranja na pitanje) se zadaju u vidu teksta ili slike. Kod pitanja sa ponuđenim odgovorima, sastavljač može da definiše veći broj odgovora od onih koja će se prikazivati pred igračem, da bi izbor ponuđenih odgovora mogao da se vrši po slučajnom principu. Sastavljač mora da označi tačan odgovor.

Sastavljač zadaje ishod davanja odgovora, odustajanja od davanja odgovora ili isticanja vremena izborom jedne od ponuđenih mogućnosti:

- ako je dat tačan odgovor:
  - igrač ostaje na istoj poziciji, a na red dolazi naredni igrač
  - igraču se dozvoljava da ponovo igra, odnosno da ne prepusti red igranja sledećem igraču
- ako je dat netačan odgovor:
  - igrač se vraća na polje na kom je prethodno bio i propušta sledeće bacanje kockice, a na red dolazi naredni igrač
  - igrač se vraća na startno polje
  - igrač se premešta na naznačeno polje i propušta sledeće bacanje kockice, a na red dolazi naredni igrač
- ako je igrač odustao od davanja odgovora, odnosno nije uneo odgovor u predviđenom vremenu
  - igrač se vraća na polje na kom je prethodno bio, a na red dolazi naredni igrač
  - igrač se premešta na naznačeno polje, a na red dolazi naredni igrač

Definisanje polja za igru se vrši na sledeći način:

- sastavljač crta jednu izlomljenu liniju u čijim su temenima polja; polja mogu da budu kružnog ili kvadratnog oblika
- sastavljač definiše sadržaj polja za igru:
  - prazno polje (podrazumevano stanje polja prilikom dodavanja na tablu)
  - polje sa pitanjem, za koje sastavljač bira težinu i oblast
  - polje sa zagonetnim pitanjem (oblast pitanja bira jedan od ostalih igrača, videti objašnjenje u nastavku)

## **Zadatak A2**

*Olimpijada Znanja - Takmičenje: Razvoj alata za sprovođenje igre takmičenja*

Igra (Takmičenje) treba da ima sledeće funkcionalnosti:

1. izbor scenarija (odnosno table) za igru, broja i imena igrača
2. izbor načina prikazivanja table ako pozadinska slika, uključujući i definisana polja, ne može da stane na ekran bez skaliranja

- a. skaliranje slike tako da cela bude vidljiva
  - b. dodavanje *scroll bar*-ova pomoću kojih igrač bira koji deo slike će biti vidljiv u datom trenutku
3. prikazivanje table i polja za igru, figura igrača, spiska igrača, imena igrača koji je na potezu i kockice za igranje; ako polje sadrži pitanje, boji se bojom koja je dodeljena oblasti iz koje je pitanje, a nijansa boje se bira u zavisnosti od težine pitanja (tamnije nijanse označavaju teža pitanja); ako polje sadrži tajno pitanje, u polju se crta "?".

Igra se izvodi na sledeći način:

1. Igrač koji je na redu "baca" kockicu, odnosno klikne mišem na nju. Kockicu ne treba animirati ali treba napraviti privid da se ona okreće tako što će se tokom nekoliko sekundi, sa početkom u trenutku "bacanja", smenjivati vrednost koju pokazuje kockica. U početku su smene brze i progresivno usporavaju kako protiče vreme predviđeno za trajanje "bacanja" kockice.
2. Figura dodeljena igraču na potezu se pomera za odgovarajući broj polja. Ako je polje na koje se figura pomera prazno, sledeći igrač dolazi na red da igra. Ako dato polje nije prazno, igraču se postavlja pitanje odgovarajuće težine i oblasti. U slučaju da se na polju nalazi tajno pitanje, onda oblast bira neki od ostalih igrača, po slučajnom izboru.
3. Pre prikazivanja pitanja, igraču se predočavaju ishodi u slučaju davanja tačnog ili netačnog odgovora, odnosno odustajanja/isticanja vremena. Pitanje se prikazuje tek nakon što igrač klikne na dugme "Spreman". Nakon toga se prikazuje pitanje i preostalo vreme za davanje odgovora. Ako igrač ne da odgovor pre isteka vremena, postupa se kao da je igrač odustao od davanja odgovora.

## DIPLOMSKI RAD

Završni rad se sastoji od samostalnog rešavanja Zadatka A1 i Zadatka A2.

## MASTER RAD

Diplomski rad se sastoji od samostalnog rešavanja Zadatka A1 i Zadatka A2, kao i sledećih dorada.

1. Predvideti posebnu kategoriju pitanja sa ponuđenim odgovorima koja omogućava sastavljaču da zada parametrizovano pitanje, a da se odgovori sastoje od para  $(X,Y)$ , gde će  $X$  figurirati u tekstu pitanja a  $Y$  u ponuđenom odgovoru. Na primer, u zadatku prepoznavanja zastava država, sastavljač pitanja treba da dostavi parove (*naziv države, slika zastave*), a parametrizovano pitanje bi bilo: "Koja od ponuđenih zastava je zastava države  $Z$ ?" gde će se tokom igre  $Z$  zameniti slučajno izabranim nazivom države.
2. Definisane polja za igru se može vršiti i na sledeći način:
  - sastavljač crta jednu ili više krivih linija (preko pozadinske slike) koje se međusobno ne presecaju niti formiraju petlje
  - sastavljač postavlja polja za igru duž definisanih krivih linija klikom levim dugmetom miša na željeno mesto na odabranoj liniji; polja se automatski uređuju prema udaljenosti od početka krive (prvo polje na krivoj je najbliže početku krive); polja ne mogu da se preklapaju
  - sastavljač definiše polje starta (gde se sve figure nalaze na početku) i polje cilja
  - ako želi, sastavljač definiše veze između nesusednih polja (polja na istoj krivoj koja nisu uzastopna ili polja na različitim krivama); automatski se proverava da li svakom putanjom može da se dođe do cilja i izdaje se upozorenje ako postoji putanja kojom ne može
3. Unos i reprodukcija zvučnih (glasovnih) pitanja, kao i poruka za uspeh, odnosno neuspeh, u odgovaranju.

4. Pravljenje dnevnika igre za svakog igrača (koja polja je obišao, koja pitanja je dobijao i kako je odgovarao)

Moguće su još neke dorade.

## **Zadatak B**

*Novčanik*: Razvoj obrazovnog softvera za pomoć u obuci rukovanja novčanicama

Cilj razvoja je obrazovni softver za simulaciju manipulisanja novcem pri kupovini proizvoda. On treba da pomogne u obučavanju aktivnostima plaćanja i provere kusura. Obučavani treba da nauči kako da što preciznije iskombinuje papirne novčanice i kovani novac da bi platio proizvod date cene i da zatim proveri vraćeni kursor. *Novčanik* se sastoji od dve aplikacije: Editora, alata kojim instruktor priprema lekcije i Trenažera, alata koji koristi obučavani.

### **Zadatak B1**

*Novčanik - Editor*: Razvoj alata za definisanje lekcija

**Alat za instruktora (*Editor*) treba da ima sledeće funkcionalnosti:**

1. unos slika apoena (papirnih novčanica i kovanog novca) i pridruživanje vrednosti
2. unos slika proizvoda i odgovarajućih cena
3. unos dozvoljenog vremena za izbor apoena pri plaćanju i vremena za brojanje kusura
4. pamćenje lekcije
5. postavljanje podrazumevane lekcije za Trenažer.

Podržati sledeće formate slika: JPG, PNG, GIF, BMP. Predvideti mogućnost promene unetih slika i pridruženih vrednosti. Uz Editor dostaviti primer datoteke sa pripremljenim slikama apoena i proizvoda sa pridruženim vrednostima.

### **Zadatak B2**

*Novčanik - Trenažer*: Razvoj alata za izvođenje lekcija i uvežbavanje

**Alat za obuku (*Trenažer*) treba da ima sledeće funkcionalnosti:**

1. učitavanje podrazumevane lekcije
2. prikaz proizvoda i cene
3. formiranje i prikaz sadržaja "novčanika", odnosno slučajnog broja istih ili različitih papirnih novčanica i kovanog novca, dovoljnih da se plati proizvod i da ostane kursor
4. uzimanje i vraćanje novčanica iz novčanika
5. plaćanje (pritiskom na dugme kada obučavani odluči)
6. vraćanje tačnog ili pogrešnog kusura
7. proveru kusura od strane obučavanog.

Program automatski na slučajan način bira proizvod i prikaže njegovu sliku i cenu. Takođe, program prikaže i slike raspoloživih papirnih i kovanih novčanica u novčaniku. Zatim se korisniku omogući da kombinuje apoene uzimanjem i vraćanjem u novčanik, sve dok ne pritisne dugme "Plaćam". Ispitaniku se saopštava da li je dobro izabrao novčanice i sugeriše mu se bolja kombinacija. Zatim program određuje i vraća kursor, ali tako da može da postoji greška, te se od obučavanog traži da proveri kursor i odredi grešku. Odgovor obučavanog treba da bude izbor jedne od tri mogućnosti: (1) kursor je tačan, (2) kursor je veći za <iznos> dinara i (3) kursor je manji za <iznos> dinara. Tačan odgovor se nagrađuje pohvalom, a netačan prati "prekorom".

## DIPLOMSKI RAD

Diplomski rad se sastoji od samostalnog rešavanja Zadatka B1 i Zadatka B2.

## MASTER RAD

Za master rad potrebne su sledeće dorade.

1. Pokazni režim. Program sam bira i prikazuje optimalnu kombinaciju apoena za prikazani proizvod date cene, a zatim prikazuje tačan kusur. Sve uz prateća tekstualna i zvučna objašnjenja.
2. Režim testiranja sa bodovanjem. Boduju se radnje plaćanja i brojanja kusura. Bodovi se upisuju u fajl, ili se preko veb-servisa upisuju u bazu podataka. Bodovanje može biti prikazano ispitaniku, a ne mora. Posebna mala aplikacija za prikaz i analizu bodova. Politika bodovanja je sledeća. Pri plaćanju, maksimalan broj bodova donosi kombinacija apoena koja daje najbližu moguću sumu novca jednaku ili veću od cene proizvoda, dok ostale kombinacije apoena donose manje bodova, prema nekom razumnom kriterijumu. I aktivnost provere kusura se boduje tako što maksimalan broj poena donosi tačan odgovor, a netačno određen iznos – umanjen broj poena proporcionalno učinjenoj greški.
3. Unos i reprodukcija zvučnih (glasovnih) pojmova (za svaki apoen), kao i poruka za uspeh, odnosno neuspeh, u rešavanju pojedinih zadataka.

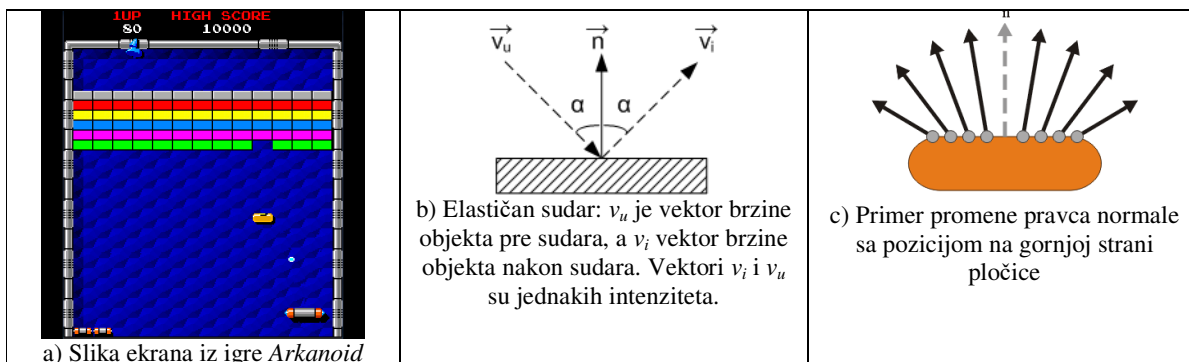
Moguće su još neke dorade.

### Zadatak C

#### *Rušenje zida: Razvoj video-igre u kojoj igrač lopticom ruši zid sastavljen od cigala*

Cilj razvoja je zabavna 2D igra za jednog igrača. Zadatak igrača je da sruši zid koji se sastoji od nekoliko slojeva cigala. Igra se odvija u oblasti koja je zatvorena sa gornje i bočnih strana, a otvorena sa donje strane. Igrač upravlja pločicom kojom odbija lopticu i mora da vodi računa da loptica ne propadne kroz donju stranu oblasti. Zadatak igrača je da lopticu (odbijanjem od pločice kojom upravlja) usmeri ka ciglama. Kada loptica pogodi ciglu, cigla nestaje. Opisana igra je inspirisana igrama sa kućnih računara iz osamdesetih i početka devedesetih godina prošlog veka. Kao ilustracija, na slici 1a prikazan je početak prvog nivoa igre *Arkanoid*, jedne od najpopularnijih igara na temu razbijanja zida.

Svi sudari loptice sa ciglama, ivicama oblasti ili pločicom koju kontroliše igrač su elastični, odnosno loptica ne menja intenzitet brzine, a ugao pod kojim loptica udara u prepreku jednak je uglu pod kojim se od prepreke odbija, što je ilustrovano na slici 1b.



Slika 1

Igrač pomera pločicu upotrebom miša. Na početku, loptica se nalazi pričvršćena za pločicu. Igra počinje nakon što igrač pritisne dugme, kada se loptica odvaja od pločice pod slučajnim uglom. Igra se završava pobedom, nakon što igrač poruši sve cigle, ili porazom, ako loptica propadne. Meri se i prikazuje tekuće vreme koje se zaustavlja u trenutku pobeđe, a postavlja na 0 u trenutku poraza.

## **DIPLOMSKI RAD**

Diplomski rad se sastoji iz sledećih dorada

1. Ugao pod kojim se loptica odbija od pločice zavisi od mesta na kojem pogodi pločicu. Ovo treba modelirati promenom ugla koji normala zaklapa sa podužnom osom pločice, kao što je prikazano na slici 1c.
2. Postojanje više "života" u igri. Propadanje loptice smanjuje broj "života" za 1. Igrač gubi igru kada broj života padne na 0.

## **MASTER RAD**

Master rad se sastoji od svih zahteva postavljenih u okviru diplomskog rada, kao i sledećih dorada.

1. Bonusi u igri. Razbijanjem svake cigle, sa određenom verovatnoćom, stvori se bonus koji pada ka dnu ekrana od mesta na kome je bila cigla. Dodirivanjem bonusa pločicom za igru, igrač aktivira dodirnut bonus, a prethodno aktivan bonus se gubi. Svaka vrsta bonusa je vizuelno jedinstvena. Predvideti sledeće bonuse:
  - a. dupliranje širine pločice za igru
  - b. trajno povećanje broja "života" za 1, ali ne poništava tekući bonus
  - c. privremeno zatvaranje donje strane oblasti, tako da loptica ne može nigde da propadne
  - d. usijanje lopte, kada lopta promeni boju u crvenu i ne odbija se od cigli prilikom rušenja
2. Uvođenje nivoa u igru. Nivo se završava kada igrač poruši ceo zid na tekućem nivou, kada se automatski prelazi na naredni nivo. Svaki nivo ima pozadinu različitu od ostalih nivoa, kao i raspored cigala (oblik zida)

Moguće su još neke dorade.

## **Zadatak D**

### *Platformer: Razvoj video-igre borbe sa protivnikom na platformama*

Ovaj domaći zadatak predstavlja proširenje zadatka rađenog na 2. laboratorijskoj vežbi i sve navedene stavke predstavljaju proširenja u odnosu na zahteve za 2. laboratorijsku vežbu. Parametre koji ne mogu da se podešavaju od strane korisnika zadati tako da igra bude interesantna (odrediti ih empirijski).

1. Platforme
  - Dodati relativno kratke platforme različitih dužina na različitim visinama na koje igrač može da skoči i na kojima ga očekuju protivnici.
  - Dodati rupe u osnovnoj platformi kroz koje igrač može da propadne i izgubi život.
2. Interakcije
  - Ako igrač skoči na glavu protivnika ubija ga i dobija poene, ako dotakne protivnika drugim putem – gubi život.
  - Igrač ima mogućnost da ispaljuje metke na protivnike ukoliko ima dovoljno municije. Meci se kreću brzinom većom od brzine igrača i pri kontaktu sa protivnikom ranjavaju ga, kada

on menja boju sa svakim primljenim metkom, a igrač dobija poene; kada primi 5 metaka protivnik gubi život, što treba ispratiti željenom animacijom, a igrač dobija dodatne poene.

### 3. Predmeti

- Dodati municiju koju igrač može da podigne sa platforme tako što je dotakne.
- Dodati novčiće koje kada igrač dotakne dobija poene. Novčiće animirati tako da blago osciluju gore-dole, kao i da se rotiraju oko vertikalne ose.
- Dodati jednostavnu zastavicu ili drugi predmet koji predstavlja kraj igre; kada igrač dotakne zastavicu kraja igre, igra se završava, ispisuje se osvojeni broj poena i nudi se da se igra ponovi ili definitivno okonča.

### 4. Protivnici

- Dodati još jednu vrstu protivnika, koji drugačije izgleda i drugačije se ponaša (na primer, puca na igrača i/ili skače).
- Ograničiti putanje po kojima se kreću (na primer, idu do kraja platforme ili nekog bloka, a onda se vraćaju).

### 5. Igrač

- Zameniti igrača gotovim slikama u svim stanjima. Preporučuje se da student koristi gotov "sprite sheet" (pogledati sliku ispod kao i linkove) koji može da nađe na Internetu. U suštini, za svako stanje u kojem se igrač nalazi treba posebna slika, a u slučaju da igrač trči treba da se smenjuju slike tako da se trčanje animira.



Primer "sprite sheet"-a

Linkovi ka mogućim "sprite sheet"-ovima:

<http://opengameart.org/content/castle-platformer>

<http://opengameart.org/content/classic-hero>

<http://opengameart.org/content/cat-fighter-sprite-sheet>

- Alternativno, zadržati crtanog igrača, ali osmisлити neki realističniji oblik i animirati ga.

### 6. Pozadina

- Dodati animirane ptice koje lete u pozadini (igrač ne može da interaguje sa njima).
- Dodati efekat paralakse, gde u pozadini postoje drugi objekti (na primer, planine, šume, pustinje) koji nisu "beskonačno" daleko tako da igrač pomeranjem može da oseti daljinu. To se može uraditi tako što se taj deo pozadine pomera nekoliko puta sporije od igrača. Pozadina ima više slojeva, gde se svaki dalji sloj pomera sporije. Koristiti bleđe palete boja na većoj udaljenosti, da se dodatno zapaža efekat daljine.

### 7. Interfejs

- Izmeniti vreme tako da se na početku prikazuje vreme za celu igru a onda se ono smanjuje. Kada istekne, igra se automatski završava. Takođe, izmeniti format prikazivanja vremena da bude "mm:ss" gde mm predstavlja broj minuta do kraja igre, a ss broj sekundi. Na primer, za 90 sekundi do kraja trebalo bi ispisati "01:30".
- Dodati da se ispisuje osvojeni broj poena.
- Dodati da se ispisuje koliko municije je ostalo igraču.
- Dodati da se ispisuje koliko je igraču ostalo života do kraja igre korišćenjem slika (ako je igraču ostalo 3 života, mogu se iscrtati 3 srca u uglu, 3 glave igrača ili staviti jedno srce/glava igrača kojem prethodi "3x", gde 3 predstavlja broj života, a x množenje).

### 8. Kamera

- Scena je šira i viša od njenog dela prikazanog kamerom.
- Implementirati pokretnu kameru koja prati igrača pri njegovom kretanju na sledeći način. Dok se igrač kreće po jednoj platformi, kamera ga prati samo horizontalnim translatorskim pomeranjem. Kada igrač promeni platformu, kamera se translatorsno pomera i po vertikalnoj osi.