Razvoj i testiranje kognitivne igre s dualnim zadatkom – praćenje središta kružnice u pokretu i *N-back* test

Seminarski rad iz kolegija “Interaktivni simulacijski sustavi”

Josip Matić, Robert Holovka  
22. Siječnja 2019.

Djelovođa: Doc. Dr. Sc. Siniša Popović

***U seminaru se opisuje razvoj i testiranje kognitivne video igre s dualnim zadatkom. Igra je podijeljena na kontinuirani i diskretni zadatak. Zadatci se mogu pokretati neovisno ili skupa, te se na temelju toga uspoređuju kognitivne performanse samog korisnika. U kontinuiranom dijelu zadatka korisnik treba pratiti središte kružnice koja je konstantno u pokretu. Korisnikova je pozicija predstavljena malenim kvadratom i cilj je zadatka pozicionirati taj kvadratić što bliže središtu kružnice. Diskretni dio zadatka predstavlja N-back test u kojem se testira korisnikova kratkotrajna memorija. Korisniku se na ekran ispisuje slijed slova, te je potrebno pamtiti određeni broj zadnjih slova gdje taj broj predstavlja parametar N.***

# Uvod

Seminar razrađuje i opisuje implementaciju kognitivne igre s dualnim zadatkom koja je nastala kao rezultat projekta za predmet *Interaktivni Simulacijski Sustavi (ISS).* Glavni razlog izrade ove igre je testiranje kognitivnih performansi igrača. Igra je implementirana u programskom jeziku *C#* uz pomoć alata *MonoGame*.

Igra se sastoji od dva zadatka, kontinuirani i diskretni. Zadaci se mogu pokretati samostalno ili simultano, te je pretpostavka da će rezultati korisnika biti lošiji ako se zadatci pokrenu istovremeno. Kasnije će se rezultatima testiranja utvrditi ova pretpostavka.

Diskretni zadatak predstavlja standardni *N–back* test, gdje se korisniku ispisuje nekakav sadržaj na ekran te ako je taj sadržaj isti onom koji se pojavio prije *N* koraka, igrač bi trebao primjetiti to i reagirati na određeni način. U psihologiji se ovaj test često koristi za testiranje kratkotrajne memorije.

U kontinuiranom zadatku imamo kružnicu koja se neprestano kreče u nasumično odabranim smjerovima. Pozicija igrača je predstavljena kvadratićem i njegov je zadatak da prati središte gibajuće kružnice. Kao rezultat se ispisuje prosječna udaljenost kvadrata od središta kružnice. Postavke igre se mogu definirati pomoću konfiguracijske datoteke koja se učitava prije pokretanje igre.

Nakon uvoda slijedi opis uloga pojedinih članova tima, u trećem odjeljku je opisana kognitivna igra i sadrži dvije podsekcije, za svaki zadatak po jednu gdje se opisuje implementacija i rezultati testiranja pojedinog zadatka. Zatim slijedi zaključak, te konačno u petom odjeljku popis korištene literature koji je poslužio u izradi ovog rada.

# Uloge pojedinih članova tima

U sveukupnim poslovima na izradi seminarskog rada te pisanju ovog izvješća, članovi tima sudjelovali su na sljedeći način:

* Josip Matić – Osnovne funkcionalnosti koje dijele oba zadatka (klasa: Game.cs), implementacija kontinuirane igre, implementacija podijeljenog ekrana (engl. *split screen*), izrada konfiguracijske datoteke (Izvješće: Kontinuirani zadatak, Zaključak)
* Robert Holovka – Implementacija *N-back* zadatka, izrada konfiguracijske datoteke (Izvješće: Sažetak, Uvod, Kognitivna igra, Diskretni zadatak i Literatura).

# Kognitivna igra

U ovom odjeljku opisuje se implementacija i rezultati testiranja ove video igre. Igra je implementirana pomoću alata *MonoGame* u programskom jeziku *C#*. *C#* je jedan od mlađih jezika, nastao 2002. godine kao sastavni dio *MS .NET Framework 1.0*. To je objektno orijentiran programski jezik i kao većina današnjih modernih jezika (C++, Java itd.) spada u jezike više programske razine. *MonoGame* je besplatni alat koji omogućuje jednostavnu izradu video igri za raznovrsne platforme i temeljen je na *Microsoft XNA*.

Da bi se omogućila testabilnost ove igre, prije samog početka iste učitava se konfiguracijska datoteka s definiranim parametrima. Konfiguracijska datoteka se sastoji od parametara koji su zajednički za oba zadatka i od još dvije skupine parametara koji se posebno definiraju za svaki pojedini zadatak.

Pod zajedničkim parametrima se može definirati koji je zadatak trenutno aktivan, te tako dobivamo da u jednom trenutku može biti aktivan samo jedan zadatak ili čak da su oba zadatka simultano aktivna. Ako su oba zadatka istovremena aktivna, također se može i definirati koji će se zadatak nalaziti na lijevoj, a koji na desnoj strani ekrana, u protivnom igra zauzima cijeli prozor. Ostali zajednički parametri predstavljaju vrijeme trajanja samog zadataka i konfiguraciju prozora u kojem se zadatak pokreće, tj. može se podesiti njegova rezolucija.

Dalje slijede podsekcije u kojima je svaki zadatak opisan zasebno i prikazani su rezultati njegova testiranja.

## Kontinuirani zadatak

Kontinuirani zadatak sastoji se od kružnice koja se kreće po ekranu jednoliko pravocrtno, nepredvidivo mijenjajući smjer u nepredvidivom pravcu, i zadatak igrača je pratiti središte kružnice što bolje može. Svrha igre je testiranje korisnikovih reakcijskih sposobnosti te koordinacije između očiju i ruku.

Sama igra se može podesiti kroz nekoliko parametara. Osim standardnog *seed* parametra koji određuje ponašanje kružnice, no ne na način tako da korisnik može zapamtiti kretanje, postoje još i parametar koji određuje brzinu kružnice, parametar koji određuje brzinu igrača u odnosu na kružnicu (npr. 150% brzine kružnice), te minimalno i maksimalno vrijeme unutar kojeg će kružnica promjeniti smjer. Ti parametri bitno utječu na težinu kontinuirane igre, pri čemu je najbitnija brzina igrača u odnosu na kružnicu.

Glavni parametar koji prikazuje performanse igrača je udaljenost središta igrača od središta kružnice. Igra konstantno mjeri kolika je navedena udaljenost (*delta* u nastavku teksta), te na kraju igre rezultate ispiše u CSV datoteku koja se zatim može otvoriti tabličnim programima poput Microsoft Excela.



**Slika 3.1.1:** Prikaz kontinuirane igre

Reakcija igrača se testira upravo kroz nagle promjene smjera kružnice. Što je duži period reakcije igrača na promjenu smjera, to će kružnica postići veću udaljenost u odnosu na igrača.

Pomoćni parametri za prikaz performansi su najveća *delta*, broj izlazaka iz kružnice te vrijeme provedeno par kružnice. To su također statističke vrijednosti koje korisnik dobiva odmah po završetku igre.

**Slika 3.1.2:** Prikaz obrađenog rezultata

Na slici 3.1.2 vidi se kako obrađeni rezultat izgleda. Zadani parametri igre su bili: vrijeme igre 90 sekundi, brzina kružnice 2px/frame, brzina igrača 150%, minimalno vrijeme prije promjene smjera 0.5 sekundi, maksimalno vrijeme nakon promjene smjera 1.5 sekundi. Na X osi se nalazi vrijeme od početka igre, dok je na y osi udaljenost igrača od središta kružnice izražena u pikselima. Srednja vrijednost udaljenosti za navedenog ispitanika je 30 piksela od središta kružnice, što je solidno ako se usporedi s polumjerom kružnice od 51 piksel.

Na grafu se također može prilično jasno vidjeti i kada se dogodila nagla promjena smjera, jer se to očituje u naglom skoku udaljenosti.

Pretpostavka je da performansa igrača pada ako se mora koncentrirati na više od jedne stvari. Uz sve parametre identične kao i prije (vrijeme igre 90 sekundi, polumjer kružnice 51 piksel, brzina kružnice 2 px/frame, brzina igrača 150% brzine kružnice, period promjene [0.5, 1.5] sekundi), te nove parametre (N = 2, prikaz samo slova, trajanje prikaza slova 4 sekunde), slika 3.1.3 pokazuje da to ne mora nužno biti slučaj:

**Slika 3.1.3:** Prikaz rezultata za obje igre paralelno

Prosječna udaljenost je 24.7 piksela. Međutim, potrebno je napomenuti kako je ispitanik u ovom konkretnom slučaju imao veću maksimalnu udaljenost nego tijekom kontinuirane igre.

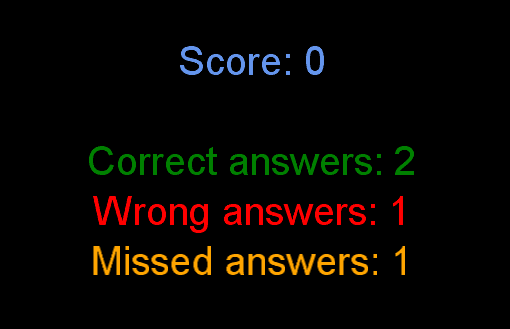
## Diskretni zadatak

Kao što je već spomenuto, diskretni zadatak je zapravo *N-back* test. Svrha *N-back* testa je testiranje korisnikove kratkotrajne memorije. U ovom zadatku, na ekran se ispisuje slijed slova, gdje se korisniku zadaje izazov da pamti zadnjih nekoliko slova. Broj slova koje je potrebno pamtiti predstavljen je parametrom *N* koji se namješta kroz konfiguracijsku datoteku u posebnom odjeljku za ovaj zadatak. Ako je trenutno slovo jednako onome koje se pojavilo prije *N* koraka, korisnik treba pravovremeno reagirati da bi postigao pozitivan rezultat. Korisnik reagira pritiskom na tipku razmak (engl. *space*). Što je parametar *N* veći, igra je izazovnija. Recimo za *N* *= 1* , korisnik treba pratiti samo uzastopno ponavljanje slova, dok za *N = 2* korisnik treba reagirati i pritisnuti tipku ako je trenutno slovo jednako onomu koje se pojavilo prije 2 koraka.

Težinu igre je također moguće definirati sadržajem koji se ispisuje korisniku na ekran. Najlakša razina predstavlja već spomenuta slova, u srednje teškoj razini slova su zamijenjena s riječima koja se međusobno uvelike razlikuju (npr. stablo, cijev, lopta...), dok se u najtežoj razini pokušava zbuniti korisnika ispisivanjem sličnih riječi (npr. mrav, mravi, mrava ...). Tranzicija između riječi/slova koja se ispisuju na ekran je implementirana pomoću tzv. *fade in/fade out* animacije čije je trajanje također moguće postaviti kroz konfiguracijsku datoteku. Taj parametar je posebice značajan ako imamo slučaj da se uzastopno ispisuje isti sadržaj i potrebno je osigurati igraču da uoči kada se sadržaj na ekranu promijenio.

Kroz konfiguracijsku datoteku također se može namjestiti koliko će dugo pojedini sadržaj biti na ekranu, te bi se pomoću tog parametra također mogla definirati i težina igre. Što je pojedina riječ/slovo kraće na ekranu, korisnik ima manje vremena za razmisliti i dati povratnu informaciju. I zadnji parametar koji se može namjestiti kroz konfiguracijsku datoteku je duljina trajanja povratne informacije. Za ovaj zadatak postoje 3 povratne informacije. Prva povratna informacija se animira zelenom bojom sadržaja i predstavlja točan odgovor, tj. korisnik je uvidio da se trenutna riječ/slovo pojavilo prije N koraka i pritisnuo je odgovarajuću tipku da potvrdi to. Druga povratna informacija animira se crvenom bojom i predstavlja pogrešan odgovor, tj. korisnik je krivo prosudio da se trenutni sadržaj pojavio prije N koraka. Zadnja povratna informacija označena je narančastom bojom sadržaja, a predstavlja situaciju kada je korisnik propustio uvidjeti da je trenutni sadržaj jednak onome koji se pojavio prije N koraka, te nije uspio pravovremeno reagirati.

Na kraju same igre korisnik dobiva povratnu informaciju, tj. rezultat i statistiku svoje igre. Konačni rezultat predstavljen je brojem točnih odgovora umanjen brojem pogrešnih i promašenih odgovora (Slika 3.1.1).



Slika 3.2.1: Ispis rezultata na kraju igre

Za potrebe testiranja u konfiguracijskoj datoteci su bili postavljeni sljedeći parametri: N = 2, trajanje igre na 3 minute, trajanje pojedinog sadržaja na ekranu 5 sekundi i težina igre na najlakšu, tj. ispisivala su se samo slova.

Zadatak je testiran na 6 subjekata koja uključuje kreatore ove igre i njihove prijatelje ili članove obitelji. Zadatak je testiran neovisno i u kombinaciji s kontinuiranim zadatkom. Također, igra je testirana na dva različita skupa podataka, tj. dvije skupine sljedova slova. Svi igrači su odigrali po jedan put diskretni zadatak samostalno i u kombinaciji s kontinuiranim zadatkom. Prvi skup podataka se pojavljivao kad se diskretni zadatak igrao samostalno, dok se drugi skup podataka pojavljivao kad se diskretni zadatak igrao u kombinaciji s kontinuiranim zadatkom. Na taj način postignuta je ispravnost samih rezultata.

Pretpostavka je bila da će kognitivne performanse igrača uvelike pasti ako igra obje igre istovremeno što i prikazuje sljedeća tablica s rezultatima testiranih subjekata.

|  |  |  |
| --- | --- | --- |
| Igrač | Rezultat samostalnog N-back testa | Rezultat N-back testa u kombinaciji s kontinuiranim zadatkom |
| 1. | Rezultat: 11 / 13 ( 85 %)  Točni odgovori: 13  Pogrešni odgovori: 2  Promašeni odgovori: 0 | Rezultat: 7 / 11 ( 64%)  Točni odgovori: 9  Pogrešni odgovori: 0  Promašeni odgovori: 2 |
| 2. | Rezultat: 12 / 16 ( 75% )  Točni odgovori: 15  Pogrešni odgovori: 3  Promašeni odgovori: 1 | Rezultat: 6 / 11 ( 55% )  Točni odgovori: 10  Pogrešni odgovori: 3  Promašeni odgovori: 1 |
| 3. | Rezultat: 10 / 13 ( 77% )  Točni odgovori: 12  Pogrešni odgovori: 1  Promašeni odgovori: 1 | Rezultat: 7 / 12 ( 58% )  Točni odgovori: 10  Pogrešni odgovori: 1  Promašeni odgovori: 2 |
| 4. | Rezultat: 10 / 12 ( 83 %)  Točni odgovori: 11  Pogrešni odgovori: 0  Promašeni odgovori: 1 | Rezultat: 8 / 11 ( 73% )  Točni odgovori: 10  Pogrešni odgovori: 1  Promašeni odgovori: 1 |
| 5. | Rezultat: 12 / 13 ( 92% )  Točni odgovori: 13  Pogrešni odgovori: 1  Promašeni odgovori: 0 | Rezultat: 8 / 13 ( 62 % )  Točni odgovori: 11  Pogrešni odgovori: 1  Promašeni odgovori: 2 |
| 6. | Rezultat: 13 / 14 ( 93% )  Točni odgovori: 14  Pogrešni odgovori: 1  Promašeni odgovori: 0 | Rezultat: 9 / 11 ( 82% )  Točni odgovori: 10  Pogrešni odgovori: 0  Promašeni odgovori: 1 |

Konačan rezultat se računa kao postotak koji se dobije dijeljenjem ostvarenog rezultata s največim mogućim rezultatom. Največi mogući rezultat je različit za svakog igrača jer ovisi o tome koliko je igrač brzo reagirao. Ako je igrač imao brzu reakciju na svaki odgovori više riječi mu se izgeneriralo i obrnuto (trajanje pojedine riječi na ekranu je 5 sekundi + vrijeme povratne poruke ako je bilo).

Iz prethodne tablice može se vidjeti da je prosječan rezultat samostalnog *N-back* testa jednak 84.1 %, dok u kombinaciji s kontinuiranim zadatkom iznosi 65.7% , što potvrđuje prije navedenu pretpostavku. Također se i vidi da neki igrači su više pogrešno odgovarali, dok su drugi više promašivali točne odgovore.

# Zaključak

Performanse osobe koja obavlja neki zadatak uvelike ovisi o tome koliko im je pažnja usmjerena na taj zadatak, te koliko dobro moće kompenzirati ometanja u radu. Iz testiranja praktičnog rada može se vidjeti da većini ispitanika performanse padaju kad moraju obavljati dva zadatka paralelno nego kad svaki zadatak obavljaju zasebno, što se poklapa sa raznim promatranjima.

Također možemo izvesti pretpostavku da ispitanik može poboljšati rezultate u testovima gdje se koriste dvije igre paralelno razvojem strategije igranja, što slike 3.1.2 i 3.1.3 i pokazuju. Takve strategije će smanjiti pad performansi pri prelasku sa zasebnih igara na obje igre paralelno, te u nekim slučajevima čak i poboljšati performanse. Međutim, potrebno je razviti drugačiji test koji će testirati upravo sposobnost korisnika da razvije strategiju za obavljanje više zadataka.

Daljnje istraživanje može krenuti u smjeru da se dodaje ili treći zadatak, ili da se počnu uvoditi i dodatna ometala poput raznih zvukova i slika. Dodatnim otežavanjem može se dosta dobro vidjeti kako osoba funkcionira pod stresom. Najvažnija činjenica je da se takvi testovi obavljaju u kontroliranim uvjetima, što znači da ispitivači mogu na temelju dobivenih podataka pokušati pronaći način kako da pomognu ljudima u situacijama istog tipa u „stvarnom svijetu“, tj. van ispitne sobe.

# Literatura

[1] MonoGame Documentation, MonoGame, <http://www.monogame.net/documentation/?page=main>

Datum pristupa: 15. studenog 2018.

[2] C# Guide, Microsoft, <https://docs.microsoft.com/en-us/dotnet/csharp/>

Datum pristupa: 15. studenog 2018.