



UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA
NOVI SAD



Projekat iz Distribuiranih Računarskih Sistema u Elektroenergetici - Frogger arkadna igra

Nemanja Simić PR16/2015

Dušan Nosović PR115/2015

Nina Grbić PR84/2015

Sadržaj

Uvod.....	3
Tok izradene igre	4
Python	4
Prednosti Python-a	5
Mane Python-a	5
PyQt5.....	6
Multiprocessing	6

Uvod

Za izradu projekta iz Distribuiranih Računarskih Sistema u Elektroenergetici odrađena je igricalik Frogger arkadnoj igrici u Python programskom jeziku. Pri realizaciji zadatka koristila se PyQt5 biblioteka, kao i multiprocessing.

Pravila ove igre nalažu sledeće:

- Igrač (žaba) inicijalno ima 3 života i kreće sa dna ekrana ka gore.
- Cilj igre je da se žaba što duže preživi i sakupi više poena.
- Poeni se dobijaju tokom kretanja ka bazi, dolaska u bazu i sakupljanja muve koja se pojavljuje u random odabranoj bazi na kratak period.
- Prilikom kretanja, ukoliko žaba udari u bilo koji automobil, ona gubi život i vraća se na početak tj. dno ekrana.
- Takođe, prilikom kretanja, ukoliko žaba upadne u reku oduzima joj se život.



Tok izrađene igre

Igra nudi multiplayer na jednom računar, multiplayer na 2 računara i turnir.

Multiplayer igra se igra sa 2 igrača koji se takmiče koji će duže preživeti i sakupiti više poena. Igra sadrži tajmer koji određuje levelisanje težine toka igre. Nakon prelaska u naredni nivo, dinamički objekti na grafičkom prikazu igrača se kreću brže i dolazak do baza odnosno vrha ekrana je otežan sa svakim sledećim nivoom.

Turnir igra se odvija na sledeći način: inicijalno započinje tačno 4 igrača. Nakon završetka igre, svačiji rezultat se pamti, 2 pobednika sa najvećim osvojenim poenima prolaze u finale i automatski započinju međusobni meč u finalu. Nakon finala, igrač sa boljim rezultatom se proglašava pobednikom turnira.

Python

Nastao 1991, Python je relativno nov programski jezik. Tokom poslednjih nekoliko godina, Python se pokazuje kao prvoklasni učesnik u savremenom razvoju softvera, upravljanju infrastrukturuom i analizi podataka. Python dopušta programerima korišćenje nekoliko stilova programiranja. Objektno orijentisano, strukturalno i aspektno orijentisano programiranje stilovi su dopušteni korišćenjem Pythona te ova fleksibilnost čini Python programski jezik sve popularnijim.

Uspeh Python-a se oslanja na nekoliko prednosti koje pruža kako početnicima tako i stručnjacima: Python se lako uči. Broj funkcija u samom jeziku je skroman, pa zahteva relativno malo uloženog vremena ili napora da se naprave prvi programi. Pythonova sintaksa je dizajnirana da bude čitljiva i jednostavna. Ova jednostavnost čini Python idealnim nastavnim jezikom i omogućava početnicima da brzo napreduju. Programeri provode više vremena razmišljajući o problemu koji pokušavaju da reše, a manje vremena razmišljaju o kompleksnosti jezika ili dešifrujući kôd koji su pisali drugi.

Python se široko koristi i podržava. Python je popularan i široko korišćen, što potvrđuje njegovo visoko rangiranje u anketama kao što je Tiobe Index i veliki broj GitHub projekata koji koriste Python. Python radi na svakom većem operativnom sistemu i platformi, a i na većini slabijih. Mnoge velike biblioteke i API-ovi servisi imaju Python vezivanja ili omotače, omogućavajući Python-u da se slobodno povezuje sa tim uslugama ili direktno koristi te biblioteke. Python možda nije najbrži jezik, ali ono što mu nedostaje u brzini, nadoknađuje se u prilagodljivosti.

Prednosti Python-a

Najosnovnije korišćenje Pythona je kao jezik skriptovanja i automatizacije. Python nije samo zamena za skriptove ili batch fajlove, već se koristi za automatizaciju interakcija sa veb pretraživačima ili GUI-jevima ili za sistemsko obezbeđivanje i konfiguraciju u alatima kao što su Ansible i Salt.

Python se koristi za opšte programiranje aplikacija. I CLI i GUI aplikacije za više platformi mogu se kreirati pomoću Pythona i raspoređuju se kao samostalni izvršni programi.

Python se koristi za nauku o podacima i mašinsko obučavanje. Sofisticirana analiza podataka postala je jedno od područja IT-a koje se najbrže razvija i jedan od zvezdanih slučajeva upotrebe Pythona. Ogromna većina biblioteka korišćenih za nauku o podacima ili mašinsko obučavanje ima Python interfejs, čineći ovaj jezik najpopularnijim komandnim interfejsom visokog nivoa za biblioteke mašinskog obučavanja i druge numeričke algoritme.

Python se koristi za veb usluge i REST API-eve. Matične biblioteke Python-a i veb-okruženja nezavisnih proizvođača pružaju brze i praktične načine za kreiraju svega, od jednostavnih REST API-ja u nekoliko redova koda, do potpuno obrađenih lokacija upravljanih podacima. Najnovije verzije Python-a imaju moćnu podršku za asinhroni rad, omogućavajući sajtovima da sa pravim bibliotekama rešavaju na desetine hiljada zahteva u sekundi.

Python se koristi za metaprogramiranje. U jeziku Python sve je objekat, uključujući i Python module i same biblioteke. Ovo omogućava Python-u da radi kao visoko efikasan generator koda, omogućavajući pisanje aplikacija koje manipulišu sopstvenim funkcijama i imaju vrstu proširivosti koja bi bila teško ili nikako izvodljiva u drugim jezicima.

Python se koristi kao lepak koda. Python se često opisuje kao "jezik lepak", što znači da može da omogući zajednički rad različitih kodova (obično biblioteke sa interfejsima C jezika). Njegova upotreba u nauci o podacima i mašinskom obučavanju je takva, ali to je samo jedna inkarnacija opšte ideje.

Mane Python-a

Python je jezik visokog nivoa, tako da nije pogodan za programiranje sistema - upravljački programi uređaja ili jezgro OS-a ne dolaze u obzir.

Takođe nije idealan za situacije koje pozivaju različite binarne platforme. Mogli biste da napravite samostalnu Python aplikaciju za Windows, Mac i Linux, ali to ne bi bilo ni elegantno ni jednostavno.

Na kraju, Python nije najbolji izbor kada je u svim aspektima aplikacije brzina apsolutni prioritet. Za to je bolji C / C ++ ili drugi jezici tog kalibra.

PyQt5

PyQt je jedan od najpopularnijih Python okvira za Qt kros-platforme C++ frejmworka. PyQt je dostupan u 2 verzije: PyQt4 i PyQt5. Obe verzije mogu da se razvijaju za Python 2 i Python 3. PyQt sadrži preko 620 klasa koje pokrivaju grafički korisnički interfejs, rukovanje XML datotekama, mrežne komunikacije, SQL baze podataka, Web pretraživače i druge tehnologije koje su dostupne u PyQt-u.

Jedne od glavnih komponenti koji PyQt5 pruža su:

- QtCore modul koji sadrži jezgro non-Gui(nevezani za korisnički interfejs) klasa, uključujući event loop, Qt signal, mehanizam slotova i sl.
- QtGui modul pretežno se odnosi na klase vezane za korisnički interfejs koji se oslanja na Model-View-Controller(MVC) prezentacionu logiku.
- QtWidgets modul
- QtWebKit modul
- QGLContext modul
- QGLFormat modul

Neke od poznatijih aplikacija koje koriste PyQt su: Dropbox, Calibre, OpenLP, Orange, TortoiseHg, Eric Python IDE, fman...

Multiprocessing

multiprocessing je modul koji podržava umnožavanje procesa koji koriste API(aplikacioni programski interfejs) sličan modulu niti gde se svaki proces izvršava nezavisno. *multiprocessing* paket pruža i lokalnu i udaljenu konkurentnost, koji koristi podprocese umesto niti. Usled toga, multiprocessing omogućava programeru da u potpunosti paralelizuje rad programa u skladu sa mašinom, odnosno raspodelom opterećenja na više procesora. Pokreće se i na Unix i na Windows OS.

Multiprocessing podržava 2 tipa komunikacionih kanala između procesa:

- Queues (redovi) - dozvoljava više potrošača i proizvođača (consumers and producers). FIFO (first-in-first-out) struktura podataka. Za preuzimanje podataka iz queue-a koristiti se metoda `get()`, dok za postavljanje podataka u queue služi `put()` metoda,
- Pipes(cevi) - po default-u dvosmerna duplex komunikacija. Primer: 2 objekta koji komuniciraju pomoću `Pipe()` kanala predstavljaju krajeve "cevi". Svaki objekat koji komunicira sadrži `send()` i `recv()` metode.

Mane multiprocessing-a:

- *multiprocessing* nije najpogodniji za proste zadatke,
- Korišćenje *multiprocessing-a* povećava kompleksnost programa i otežava proces debugg-ovanja za programera.