|  |  |  |
| --- | --- | --- |
|  | UNIVERZITET U NOVOM SADU FAKULTET TEHNIČKIH NAUKA  NOVI SAD |  |

Zadatak – Donkey Kong

Distribuirani računarski sistemi u elektroenergetici

- Primenjeno softversko inženjerstvo-

Novi Sad, 20. januar 2020.

Sadržaj

¸

* ZADATAK

Napisati igricu nalik Donkey Kong igrici. Napraviti šemu platformi, sličnu kao u originalnoj

igrici, kroz koje avatari mogu da se kreću.

Uloge:

- Dva igrača, pokreću svog avatara pomoću strelica, odnosno ASDW tastera

- Kreću se levo-desno, i gore-dole uz merdevine

- Kad se popne na platformu iznad dobije se jedan bod

- Svaki igrač ima 3 života po nivou

- Na početku nivoa, nalaze se u ćoškovima, na najnižoj platformi

- Gorila:

- Kreće se nasumično levo-desno po drugoj najvišoj platformi

- Baca burad koja padaju vertikalno dole

- Na početku nivoa, nalaze se na sredini

- Princeza:

- Stoji na najvišoj platformi i maše rukama

Pravila:

- Igra se beskonačno nivoa

- Ukoliko bure ili gorila dodirnu igrača, on gubi jedan život

- Nakon svakog nivoa burad brže padaju

- Za prelaz na sledeći nivo potrebno je popeti se do Princeze

- Pobednik je igrač koji najduže ostane u igri

- Igrica se završava kada svi igrači izgube sve živote

Za realizaciju koristiti Python3, PyQt5, multiprocessing.

Raditi u timovima od 4 člana.

Napisati dokumentaciju u kojoj opisuje opšti rad aplikacije i rezimirati prednosti i mane

korišćenja Python jezika, PyQt5 okvira i paralelizacije rada, konkretno multiprocessing

biblioteke.

* OPŠTI RAD APLIKACIJE

Prilikom pokretanja aplikacije prikazuje se početni prozor (Slika 1.) za odabir tipa igre.



Slika 1.

Biranjem između opcija 1 PLAYER GAME i 2 PLAYER GAME započinjemo sa igrom, sa jednim, odnosno dva igrača. Treća opcija QUIT služi za izlaz iz igrice.

Pri odabiru jedne od prve dve opcije, otvara nam se igrica koja izgleda ovako:



Slika 2.

Pozicija Maria i Luigia je na prvom spratu, gorile na petom i pozicija princeze na poslednjem spratu. Gorila baca burad koje Mario mora da izbegava u suprotnom gubi jedan zivot. Cilj je da jedan od igraca stigne do princeze i spasi je,kada se to dogodi prelazi se na sledeci nivo gde burad padaju brze i time otezavaju igru igracima,na svakom sledecem nivou burad sve brze i brze padaju. Igraci takodje gube zivot prilikom dodira sa gorilom i sa vatrom koja se pojavljuje na *random* mestima na tabli.

|  |  |  |
| --- | --- | --- |
| Mario |  | Kretanje po spratovima levo-desno i gore-dole  Kontrole:  strelice |
| Princeza |  | Stoji na poslednjem spratu i mase rukama |
| Luigi |  | Kretanje po spratovima levo-desno i gore-dole  Kontrole:  W,A,S,D |
| Gorila |  | Kretanje levo desno na pretposlednjem spratu, baca burad |
| Negativna neočekivana sila |  | Pojavljuje se posle 2 sekunde na random poziciji koju menja nakon 5 sekundi,oduzima zivot igracu prilikom kontakta sa njim |
| Burad |  | Burad koja padaju vertikalno dole i oduzimaju poene igracima ako ih pogode |
| Dugme za povratak u meni |  | Prikazan u desnom uglu ekrana |

Da bi se igralo, prvo je potrebno pokrenuti Menu.py fajl koji u sebi sadrži klasu Menu.



Odabirom između ove dve opcije, pokrećemo igricu sa jednim ili dva karaktera.

|  |  |
| --- | --- |
| label1Player | Klikom na sliku (QLabel) , pokreće se funkcija one\_player\_on\_click, koji prosledjuje 1 kao argument broja igrača u igri (MainWindow ) |
| label2Player | Klikom na sliku (QLabel), pokreće se funkcija two\_players\_on\_click, koji prosledjuje 2 kao argument broja igrača u igri (MainWindow) |
| button4 | Klikom na dugme (QPushButton) poziva se funkcija quit\_on\_click, nakon čega se Menu prozor zatvara |

|  |  |
| --- | --- |
| \_\_init\_\_ , \_\_init\_ui\_\_ | Konstruktori |
| keyPressEvent, keyReleaseEvent | Funkcije koje preuzimaju informacije o pritisnutom ili pustenom dugmetu, koji utiču na kretanje karaktera |
| \_\_update\_position\_\_, \_\_update\_position2\_\_ | Kretanje karaktera za svakog igrača ponaosob |
| movePrinces | Funkcija u kojoj je odradjeno princezino mahanje rukama |
| newLevel | Funkcija za prelazak na sledeci nivo i ubrzavanje paa buradi |
| moveDonkeyKong | Funkcija za kretanje gorile |
| moveBarrels | Funkcija za generisanje buradi, regulisanje ucestanosti izbacivanja |

* PREDNOSTI I MANE

Za razvoj ove igrice korišten je Python programski jezik, PyQt5 framework i multiprocess biblioteka za paralelizaciju rada.

1. Python programski jezik je interpretivni „visoki“ programski jezik koji podržava više tipova paradigmi, kao što su: objektno-orijentisana, imperativna, funkcionalana, proceduralna. Korišćenje Pythona nam je mnogo olakšava sam razvoj igrice, jer automatski vodi računa o stvarima kao što su memory managment, o tipovima podataka, lako skaliranje aplikacija, podržava cross-platform programiranje, itd. Loše strane Python-a je pre svega što je Python spor, ima ograničenja kada je u pitanju pristup bazama podataka, gotovo je nemoguće napraviti kvalitetnu 3D grafiku, i nije najbolje za rešavanje zadataka koji zahtevaju optimizaciju korištenja memorije.
2. Framework koji je korišten za izradu ove igrice je PyQt5. PyQt5 je framework pisan u programskom jeziku C++. Omogućava razvoj cross-platform GUI-a (Graphical User Interface). Sadrži oko 440 klasa i preko 6000 funkcija i metoda. Jedan od najvećih problema ovog frameworka je što je npr. na Windows platformi Qt aplikacija samo jedan prozor, bez kontrola tako da je otežano debagovanje.
3. Multiprocessing biblioteka, koja nam pruža kako lokalnu tako i distribuiranu konkurenciju. Zamenujući Global Interpreter Lock (mehanizam u Python-u koji garantuje da samo jedan thread može istovremeno da izvrašava bytecode) koristeći podprocese umesto thread-ova. Zahvaljujući ovom modulu (modul je fajl u Python- u sa ekstenzijom .py koji sadrži definicije i izraze) koji dozvoljava programeru da u potpunosti iskoristi procesore na datoj mašini i samim tim paralelizuje izvršavanje zadataka. Biblioteka se koristi i na Windows i na Unix platformama. Problem koji se javlja prilikom korištenja multiprocess biblioteke umesto thread-va je što se koriste procesi koji svaki ima zasebni memorijski prostor što otežava deljenje objekata i resursa između procesa, dok korištenjem thread-ova koristimo deljeni memorijski prostor i ovaj problem ne postoji. Kod thread-ova moramo zato koristiti neke od mehanizama zaključavanja kako ne bi više niti pisalo na istu memorijsku lokaciju, što bi dovelo do nekonzistentnog stanja prilikom izvršavanja aplikacije.

Za izradu projekta korišćeni su sledeći mehanizmi:

* Python programski jezik
* PyQt5 biblioteka
* Multithreading