



Andrija Jovanović RA117/2016

Denis Liliom RA31/2016

Strahinja Lončarski RA237/2016

Logičko projektovanje računarskih sistema 2

T-Rex, Run!

Mentor: Miloš Subotić



Sadržaj

Zadatak.....	1
Originalna realizacija	1
Implementacija igre	1
Sprite	1
Interni format sprite-a	1
Animacija.....	2
Smena dana i noći	2
Generisanje sprite-ova	2
Gameplay	3
Reference	4



Zadatak

Zadatak je bio implementacija T-Rex Run! igre na E2LP platformi.

Originalna realizacija

Igra je realizovana od strane Google kompanije u okviru Chromium projekta kao zabava za korisnike koji su privremeno ostali bez internet konekcije. Pisana je u JavaScript-u. Moguće je igrati na tastaturi gde strelica na gore kao i space predstavljaju skok, dok strelica na dole predstavlja saginjanje. Moguće je i igrati pomoću ekrana osetljivog na dodir, gde je pritisak na ekran skok.

Implementacija igre

Sprite

Sprite je osnovni grafički element koji se prikazuje na ekranu i manipuliše u toku igre. Na ekranu sprite je predstavljen pravougaonikom piksela, svaki od kojih može da bude obojen u jednu od 16 boja iz palete. Taj pravougaonik može biti proizvoljne širine i visine, ali mu širina mora biti deljiva sa 8 (zbog implementacionih detalja).

U okviru programa, svaki sprite je predstavljen strukturom `Sprite` koja u sebi sadrži podatke o njegovoj širini i visini, kao i niz boja za svaki od piksela koji čine taj sprite, skladištenim u internom formatu. Funkcijom `show_sprite()` se on prikazuje na zadatim koordinatama na ekranu, a funkcijom `clear_sprite()` se briše sa ekrana. Funkcije za rad sa sprite-ovima trebaju samo da isčitaju njihov niz sa sadržajem i upišu ga u VGA grafičku memoriju preko FSL-a na adrese koje odgovaraju željenim lokacijama na ekranu. Funkcionalnost FSL-a je opisana u dokumentaciji od prethodnih projekata.

Interni format sprite-a

Svaki piksel ima jednu od 16 mogućih boja iz palete. On se predstavlja brojem od 0 do 15 koji označava indeks te boje unutar palete. Sprite svoje boje čuva kao niz 32-bitnih brojeva. Svaki od tih brojeva je kompresovan sadržaj boja 8 uzastopnih piksela, a svaki nibble (polovina bajta) broja je boja po jednog od tih piksela. Ovo je ujedno i format u kojem se podaci o pikselima čuvaju unutar grafičke memorije VGA, tako da nije potrebno vršiti nikakvu obradu ili promenu formata pre prebacivanja.

Visina sprite-a je uvek izražena u pikselima, dok je njegova širina 8 puta manja od širine u pikselima i upravo se odnosi na ove 32-bitne brojeve internog formata.





Animacija

Animacija nekog entiteta se postiže smenjivanjem između nekoliko sprite-ova, svaki od kojih predstavlja taj entitet u jednoj od faza njegove animacije. Potrebno je napraviti po sprite za svaku od tih faza i u okviru entiteta čuvati podatak o tome koji je od tih sprite-ova trenutno aktivan.

Na osnovu neke logike će se tokom igre menjati trenutno aktivan sprite, i kada se sledeći put pozove `show_sprite()`, izgled entiteta na ekranu će se prilagoditi toj izmeni i postiće se efekat animacije. Neke animacije se periodično smenjuju (npr. mahanje krila kod ptica), dok se neke dešavaju samo nakon određenih događaja (npr. promena očiju dinosaurusu nakon smrti), ali oba slučaja se postižu promenom aktivnog sprite-a.

Smena dana i noći

Da bi se postigao efekat promene dana i noći koristi se osobina da sprite-ovi znaju samo indekse boja unutar palete, a ne njihove apsolutne vrednosti. Dovoljno je promeniti vrednost boja unutar palete i sve prikazano na ekranu će se prilagoditi tim izmenama bez ponovnog crtanja i bez potrebe da se prave odvojeni set sprite-ova za tamni i svetli mod. Za to postoje funkcije `day_mode()` i `night_mode()` čiji se pozivi periodično smenjuju.

Generisanje sprite-ova

Za generisanje sprite-ova napravljena je Python skripta koja png sliku pretvara u niz koji se može direktno koristiti u kodu. Dimenzije te slike treba da budu iste kao željene dimenzije sprite-a na ekranu, a paleta dozvoljenih boja je definisana unutar same skripte i detaljnije opisana u njenoj dokumentaciji. Za crtanje slika koristili smo Paint.NET program pošto je jednostavan za upotrebu i praktičan za crtanje piksel po piksel, ali za skriptu nije važno kojim je programom slika izgenerisana.

Izlaz skripte je C kod kojim se definiše niz boja jednog sprite-a, dok se struktura `Sprite` mora samostalno napraviti i popuniti.





Gameplay

Poenta igre je da se izbegavaju prepreke i preživi što duže. Igra nema kraj već traje sve dok igrač ne umre. Dinosaur se kontroliše džojstikom. Pritiskom na gornji taster on skače i visina skoka zavisi od vremena držanja tastera, a pritiskom na srednje dugme se sagne. Vremenom se brzina kretanja dinosaurus ubrzava (do određene granice), i što se on brže kreće to brže skuplja poene.

Zbog lakšeg testiranja uveden je "cheat" u kojem se ne proveravaju kolizije i dinosaur ne može da umre. No collision mode se pali pritiskom na desni taster na džojstiku a gasi pritiskom na levi. Na ekranu je naznačeno kada je no collision mode aktivan.





Reference

Github link projekta: <http://www.github.com/LiliFakultet/TREX>

YouTube link: <https://www.youtube.com/watch?v=Ske-izDszqQ>

Literatura: <http://www.rt-rk.uns.ac.rs/predmeti/e2/lprs-2-logi%C4%8Dko-projektovanje-ra%C4%8Dunarskih-sistema-2>

