

Projekti iz predmeta Računarska elektronika

Aleksandra Lekić

14. april 2016

Za svaki od zadataka potrebno je dostaviti:

- Folder sa kodom programa koji zadovoljava traženu funkcionalnost. Ukoliko program generiše izlazne fajlove, potrebno je i njih priložiti.
- Izveštaj u kome se opisuje funkcija zadatka i kako je sam program realizovan. Na kraju izveštaja potrebno je dodati ceo kod projekta.

Izveštaj i programe sačuvati u folderu sa imenom REx, gde je x broj projekta. Folder u vidu arhive poslati do 31.05.2016. mejlom sa *subject-om* REx_projekat na mejl adresu:

lekic.aleksandra@etf.rs

Odrbrana projekata će se organizovati po rasporedu koji će biti naknadno objavljen 03.06.2016.

PGMA format slike

PGMA je *grayscale* ASCII format slike čiji je zapis:

```
P2
# pepper.ascii.pgm created by PGMA_IO::PGMA_WRITE.
256 256
255
26 47 49 43 44 44 46 41 43 42 43 43
42 42 41 39 44 39 42 45 46 42 39 39
36 39 37 37 35 41 37 35 33 34 33 37
...
```

Prvi red čini "magični broj" koji je u slučaju PGMA formata uvek P2. Zatim sledi karakter za novi red 0x0A i red koji počinje sa #. Ovaj red predstavlja komentar. U sledećem redu je data širina slike odvojena sa dva razmaka i visina slike. Četvrti red predstavlja maksimalnu vrednost piksela slike, što je u slučaju 8-bitne slike 255. Od petog reda nadalje su date decimalne vrednosti piksela međusobno odvojene razmakom ili sa razmakom i znakom za novi red.

PGM slika se može prikazati korišćenjem programa *IrfanView* koji se može naći na linku: <http://www.irfanview.com/>. Detaljnije uputstvo je dato u datoteci *Obrada_slike* pod nazivom *pgm_files.pdf* u okviru fajlova za izradu projekta. Dato je i nekoliko slika PGMA formata koje se mogu koristiti za testiranje projekata.



Primer slike.

Na sajtu <http://people.sc.fsu.edu/~jburkardt/data/pgma/pgma.html> se može naći još primera slika, pri čemu ih je za upotrebu u programu potrebno dodatno umanjiti korišćenjem programa *IrfanView*.

U naredna tri projekta potrebno je napraviti program koji po pokretanju pita za ime slike koju je potrebno obraditi. Nakon izvršene obrade slike, pita se za naziv nove slike i upisuju se obrađeni podaci.

1. projekat

Napisati program koji pravi sliku duplo manjih dimenzija od pročitane slike decimacijom. Naime, "preskakanjem" svakog drugog piksela dobija se slika duplo manjih dimenzija od početne.

2. projekat

Napisati program koji pravi sliku duplo većih dimenzija interpolacijom. Između svaka dva piksela dodaje se jedan novi (i po širini i po visini) čija je vrednost jednaka aritmetičkoj sredini vrednosti susedna dva piksela.

3. projekat

Napisati program koji pravi sliku koja nastaje poređenjem ulazne slike sa pragom $th = 128$, tako što sve vrednosti koje su veće od th postavi na 255 (belu boju), a manje na 0 (crnu boju).

Šifrovanje

4. projekat

Napisati program koji sa standardnog ulaza čita broj n koji predstavlja pomeraj prilikom primene Cezarovog kodiranja. Napisati funkciju koja po unosu pomeraja mapira tipke tastature tako da se prilikom kucanja poruke na standardnom izlazu prikazuje šifrovana poruka.

Cezarovo šifrovanje preslikava slovo u slovo pomereno n mesta. Na primer za $n = 3$: $A \rightarrow D$, $B \rightarrow E$, $C \rightarrow F$...

5. projekat

Napisati program koji kodira tekst koji se kuca na tastaturi *Playfair*-ovom šifrom. Prilikom kucanja poruke na standardnom izlazu se prikazuje šifrovana poruka.

Playfair-ovo šifrovanje zamenjuje grupu od dva znaka abecede grupom od dva znaka po sledećem algoritmu. Konstruiše se 5×5 matrica slova koja će u našem zadatku biti:

```
P L A Y F
I J R B C D
E G H K M
N O Q S T
U V W X Z,
```

a zatim se podeli tekst na blokove od po dva slova (bez blanko znakova) koji se koduju na tri moguća načina:

- Slova se nalaze u istom redu. Tada ih zamenimo sa slovima koja se nalaze za jedno mesto udesno (ciklično). Npr. $EH \rightarrow GK$, $ST \rightarrow TN$, $FP \rightarrow PL$.
- Slova se nalaze u istoj koloni. Tada ih zamenimo sa slovima koja se nalaze za jedno mesto ispod (ciklično). Npr. $OV \rightarrow VL$, $KY \rightarrow SC$, $PI \rightarrow IE$.
- U protivnom, pogledamo kvadrat koji određuju ta dva slova, pa ih zamenimo s preostala dva vrha tog kvadrata. Redosled je određen tako da prvo dolazi ono slovo koje se nalazi u istom redu kao prvo slovo u polaznom bloku. Npr. $OC \rightarrow SR$, $RK \rightarrow CG$, $PD \rightarrow FI$.

Znakovi I i J se poistovećuju.

6. projekat

Napisati program koji kodira tekst koji se kuca na tastaturi *Vigenèr*-ovom šifrom. Prilikom kucanja poruke na standardnom izlazu se prikazuje šifrovana poruka. Prilikom pokretanja teksta se pita prvo za ključnu reč. Ključna reč treba da bude pisana velikim slovima, kao i sam tekst. Nakon toga se po kucanju teksta kodira reč po reč i ispisuje na standardni izlaz.

Primer kodiranja je:

Otvoreni tekst: ATTACKATDAWN

Ključ: LEMONLEMONLE

Šifrat: LXFOPVEFRNHR.

Naime, otvoreni tekst se sabira po modulu 26 sa ključem. Ukoliko je ključ duži od teksta, sabira se onoliko znakova koliko je dugačka reč teksta. Ako je reč duža od ključa, ključ se ponavlja koliko je puta potrebno kao u primeru iznad. Formula za računanje vrednosti kodirane reči je:

$$S_i \equiv O_i + K_i \pmod{26},$$

gde je O_i , i -to slovo reči, K_i ključa, a S_i slovo kodirane reči.

Analiza ulaznog fajla

7. projekat

Napisati program koji analizira ulazni tekstualni fajl i na osnovu njegovog sadržaja formira odgovarajući ispis na ekranu. Ulazni fajl je oblika:

[R, G, B] tekst pauza

Prvi karakter označava boju teksta *R-red*, *G-green* i *B-blue*, **tekst** je tekst koji treba da se ispiše u zadatoj boji, a pauza je vreme u sekundama koliko treba da se sačeka dok se ne ispiše sledeća poruka u novom redu.

8. projekat

Napisati program koji će analizirati ulazni fajl. Svaki red u ulaznom fajlu je sekvenca karaktera: **tekst** [pauza], gde je **tekst** - sekvenca karaktera (ne sme da sadrži blanko znak), **pauza** - vreme u sekundama nakon kojeg treba obrisati poruku sa ekrana i ispisati novu. Na osnovu sadržaja fajla formira se odgovarajući ispis na ekranu. Na kraju izvršavanja programa potrebno je ispisati njegovo vreme izvršavanja.

9. projekat

Napisati program koji će analizirati ulazni fajl. Svaki red u ulaznom fajlu je sekvenca karaktera: x_0 y_0 x_1 y_1 boja, gde su x_0 , y_0 koordinate gornjeg desnog ugla kvadrata, x_1 , y_1 koordinate donjeg desnog ugla kvadrata, boja - boja kvadrata. Na osnovu sadržaja fajla iscrtava se odgovarajući broj kvadrata na ekranu.

Kontrola kursora

10. projekat

Napraviti *Bouncing ball* igricu sa pokretnim "zidom" od koji se odbija loptica (u vidu nekog znaka) sa donje strane. "Zid" kontrolisati levo desno sa dva tastera na tastaturi, unapred definisana. Igrica se završava kada "loptica" nema u šta da uradi sa donje strane ekrana.

11. projekat

Koristeci tastere na tastaturi pomerati kursor u cetiri pravca. Pritiskom tastera B na tastaturi cursor se pomera na donju ivicu po sredini prozora, tastera U na gornju ivicu po sredini, L levo po sredini i R desno po sredini. Pritiskom tastera ENTER završava se izvršavanje programa. Sve pritisnute tastere je potrebno čuvati u memoriji i po izvršavanju programa sve akcije zapisati u odgovarajući tekstualni log fajl.

12. projekat

Koristeci strelice na tastaturi definisati iscrtavanje vertikalnih linija. Naime, u programu zadati početnu vrednost kursora. Pritiskom strelice na gore ili na dole na tastaturi se iscrtava horizontalna linija (ili linija od bloka karaktera DBh) pomerena za 20 odnosno -20 u odnosu na početnu tačku. Analogno, pritiskom strelice ulevo/udesno se iscrtava vertikalna linija pomerena -20/+20 ulevo/udesno od tekuće lokacije. Pritiskom tastera ENTER završava se izvršavanje programa.

13. projekat

Napisati program koji iscrtava mali kvadrat na ekranu korišćenjem nekoliko blokova (ASCII kod DBh) u boji koja se zadaje na početku programa na standardnom ulazu. Moguće je zadati tri boje unosom sledećih karaktera: R-*red*, G-*green* i B-*blue*. Pomerati kvadrat po ekranu na osnovu random generisanih smerova. Koristiti fiksno kašnjenje od 50 ms.

Promena formata fajla

M3U format plejliste ima format:

```
#EXTM3U
#EXTINF:213,Evanescence - Breathe No More
E:\Muzika\Evanescence Discography\Evanescence - Anywhere But Home\Evanescence -
Breathe No More.mp3
#EXTINF:283,Evanescence - Bring Me To Life
E:\Muzika\Evanescence Discography\Evanescence - Anywhere But Home\Evanescence -
Bring Me To Life.mp3,
```

gde je #EXTM3U ključna reč kojom počinje plejlita. Zatim slede redovi sa numerama koje počinju ključnom reči #EXTINF:, a potom sledi podatak o trajanju u sekundama i naziv numere. U sledećem redu je data putanja do fajla numere.

PLS format plejliste ima format:

```
[playlist]
NumberOfEntries=50
File1=nesto-nesto.mp3
Title1=nesto- nesto.mp3
Length1=1
File2=E:\Muzika\Dream Theater\Falling Into Infinity\01_New Millennium.mp3
Title2=Dream Theater - New Millennium
Length2=500,
```

gde je [playlist] ključna reč kojom počinje plejlita, a drugi red je broj numera u plejliti. Zatim slede redovi sa numerama koje su date u tri reda, gde prvi predstavlja putanju do fajla, drugi ime numere, a treći trajanje u sekundama.

U folderu Plejlita su dati primeri plejliste u oba formata.

14. projekat

Napisati program koji prevodi ulaznu M3U plejlistu u PLS plejlistu. Potrebno je da se po pokretanju programa traži ime plejliste i da se formira plejlita drugog formata sa istim imenom, ali drugom ekstenzijom.

15. projekat

Napisati program koji prevodi ulaznu PLS plejlistu u M3U plejlistu. Potrebno je da se po pokretanju programa traži ime plejliste i da se formira plejlita drugog formata sa istim imenom, ali drugom ekstenzijom.