

REGE – program za generisanje rebusa

Napravljen je program koji od bilo koje rečenice ili riječi na srpskom jeziku pravi rebus. Koriste se pravila za rješavanje koja su napravili enigmatičari. Tekst se rastavlja na riječi koje se mogu predstaviti slikama. Pri tom se jedna riječ može predstaviti sa više slika, a slučajnim izborom se određuje koja će slika biti upotrebljiva. Fond slika se povećava na jednostavan način. Dobijeni rebusi se mogu odštampati direktno iz programa ili snimiti u .bmp formatu i kasnije obraditi (u nekom drugom programu).

Uvod

Rebusi su zagonetke u slikama. Međutim, prva zagonetka vezana za rebuse je kako oni nastaju. Prave ih ljudi specijalizovani za taj posao koji pored iskustva posjeduju i izvestan talenat, ali i dokoličari zaljubljeni u enigmatiku. Većinom ih prave ljudi iz dosade, a služe drugim ljudima za ubijanje iste. Postupak kojim rebusi nastaju je naporan, pogotovo ako kreator nema bar neku vrstu inspiracije. Iz tog razloga malo ih se objavljuje, a pošto se teško dolazi do novog rebusa često se ponovo objavljuju. Samim tim nisu originalni, a originalnost najviše privlači rješavaoce rebusa.

Postoji način po kojem bi se od svakog teksta došlo do rebusa, ali on oduzima dosta vremena. Otuda i ideja da računar, umjesto čovjeka, obavlja taj posao. Program koji bi to radio treba da omogući korisniku da za svaku rečenicu ili riječ dobije rebus čije bi rješenje bila upisana rečenica. Na taj način bi se lako, brzo i bez napora dolazilo do novih, atraktivnih rebusa.

Ovaj program generiše rebus na osnovu zadatog teksta. Rađen je u Borland Delphi-ju, a programski interfejs je urađen na srpskom jeziku. Korisnik može izabrati pismo koje će se koristiti u izradi rebusa (ćirilica ili latinica). Treba spomenuti da je na srpskom jeziku jednostavnije napraviti rebus nego na nekom drugom, pošto svakom slovu odgovara samo jedan glas i obrnuto.

*Radmila Tomić (1981),
Banja Luka, Slavka
Rodića 2B, učenica 3.
razreda Gimnazije u
Banjoj Luci*

Opis rada programa

Opšte napomene

Program pravi rebus tako što rastavlja unjeti tekst na riječi koje se mogu predstaviti slikama. Riječi koje stoje na raspolaganju se nalaze u bazi (format .dbf), a slike u posebnom poddirektorijumu. Postupak za dodavanje novih slika se sastoji u tome da se slike (u .wmf formatu) snime u poseban direktorijum, a zatim se u bazu upišu nazivi slika i potrebni padežni oblici.

Unjeti tekst može sadržati velika, odnosno mala slova i interpunkcijske znake. Jedino ograničenje se odnosi na dužinu teksta (tj. generisanog rebusa). Ako rebus sadrži više slika nego što je moguće prikazati, program će upozoriti korisnika da upiše kraći tekst. Sav tekst se na početku pretvara u mala slova radi jednostavnije pretrage baze raspoloživih reči. Zatim se zadati tekst čisti od svih znakova koja nisu slova, budući da se oni ne mogu predstaviti slikama. Da bi rješavalac znao da postoje takvi znakovi, oni se, kao i dužine riječi, pišu iznad rebusa (npr. za frazu *da li ovo radi?* iznad rebusa će pisati _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ ?). Prije nego što se obrišu praznine i tekst postane jedna 'riječ' zapamte se upisane riječi da se one ne bi pojavile u rješenju. (Na primjer, kada bi kao rješenje za unjetu riječ *telefon* dobili sliku telefona to nikako ne bi mogli nazvati rebusom.) Tek nakon toga se prelazi na generisanje rebusa.

Osnovna struktura

Sledeći kod predstavlja metod po kojem se generiše rebus:

BEGIN

REPEAT

Ako se na početku tražene riječi (riječ na osnovu koje se generiše rebus) nalazi *ne*, *na*, *pod*, *o*, *kraj* ili *pored* (npr. ako je tražena riječ *nekome*, na početku je *ne*)

onda najbolja riječ postaje ona riječ koja je nađena (*ne*, *na*, *pod*, *o*, *kraj*, *pored*);

početak baze

REPEAT

Riječ iz baze dobija vrijednost na osnovu toga koliko slova je isto, a koliko različito sa traženom riječi; (npr. riječ *šine* ima dva različita (*š* i *i*) i dva ista (*n* i *e*) slova sa riječi *nekome*) ako riječ ima veću vrijednost od trenutno najbolje ona postaje najbolja;

(trenutno najbolja riječ je *ne* pošto se oba slova poklapaju sa traženom)

UNTIL kraj baze; (provjere se sve riječi iz baze)

u rješenje se upisuju karakteristike najbolje riječi (koliko slova se treba oduzeti ili dodati i koja su to da bi se najbolja riječ poklopila sa traženom);

rečenica se skraćuje za dio koji je već predstavljen;

(od *nekome* se oduzima *ne* i ostaje *kome*; sada je tražena riječ *kome*)

UNTIL tražena riječ='';

crta se rebus po modelu iz rješenja;

END.

Enigmatski trikovi kod rebusa i njihovo rješenje u programu

Šta je ustvari dobar rebus i kako se zna koji je od dva rebusa bolji? Onaj koji sa manje slika predstavi rečenicu po pravilu je bolji. Obično se ljudi oduševavaju kad otkriju da je komplikovana rečenica predstavljena sa malo slika. Da bi se to postiglo koriste se slova i apostrofi. Na primjer, ako najbolja riječ koja treba da zamjeni traženu ima samo jedno zajedničko slovo sa njom, onda se ispisuje samo to zajedničko slovo (ljepše izgleda slovo od slike sa gomilom apostrofa). Apostrofi imaju suprotan zadatak. Ako bi se neka riječ savršeno uklopila u naš rebus, samo da nema prvog slova, onada ćemo jednim apostrofom ispred slike naglasiti da se prvo slovo treba izostaviti. Uopšte, apostrofi stoje ispred ili iza slike u zavisnosti od toga da li se treba izostaviti prvo ili poslednje slovo riječi koja je predstavljena slikom (koliko ima apostrofa toliko se slova treba izostaviti). Postoji i mogućnost da se dvije riječi, ona koju tražimo i ona koju imamo na raspolaganju, razlikuju u samo jednom slovu. U tom slučaju se treba dopisati slovo i staviti apostrof, a radi se o slovima unutar riječi. Najljepše rješenje za takav problem je da se ispod slike naznači koje se slovo treba zamjeniti

Primjer: vruće = vreće (e=u)



Da bi se rebus sastojao od malo slika, potrebno je neke riječi, odnosno dijelove riječi, predstaviti na neki drugi način. Zato u ovom programu postoje standardne zamjene za slike.

Ako se *ne* sadrži u traženoj riječi i ne postoji riječ koja bi se bolje uklopila onda će se precrtati sledeća slika.

Primjer: nekome



Prva slika predstavlja rješenje sa opcijom *ne*, a druga je rješenje bez te opcije.

Drugi trik koji koriste enigmatičari je sledeći. Prijedlog *u* se predstavlja tako što se jedna slika stavi u drugu. Ovo je programski teško uraditi pošto program ne prepoznaje svaku sliku, pa da zna gdje bi mogao umetnuti drugu. Ako bi se uvijek crtalo na istom mjestu moglo bi se desiti da se slike preklope i da postane nejasno šta one predstavljaju. Zbog toga je ta opcija realizovana samo djelimično, odnosno samo za slova.

Primjer: ruka i sud



Slično ovoj postoji i konstrukcija dva slova o trećem; ova opcija takođe nije realizovana za slike.

Primjer: televizor i život



Prijedlozi *na*, *pod*, *kraj* i *pored* se predstavljaju tako što se slike postavljaju jedna na, pod, kraj ili pored druge. Takvi rebusi se mogu tumačiti na više načina (na primjer ista slika je rješenje za *pero pod stolom* i *sto na peru*). Ovdje se pojavio problem pošto prjedlozi sa sobom povlače i određene padeže. Znači slika stola se više ne čita kao sto već kao stolu, stolom ili stola. Ovaj problem je prevaziđen tako što se u bazi, pored svake riječi, nalaze i određeni padežni oblici (lokativ, instrumental, genitiv). Po potrebi traže se određeni oblik riječi.

Primjer: na podu, snaha



Primjer: s'kraja i sporedan



Pravila koja se koriste pri rješavanju rebusa

ZNAK	ŠTA ZNAČI
Slovo	Slovo se piše ispred ili iz riječi (koja je predstavljena slikom)
Apostrof	Broj apostrofa predstavlja broj slova koja se trebaju oduzeti sa početka ili kraja riječi
=	Zamjenjuju se slova (ako ispod slike ruke stoji r=m rješenje je muka)
RIJEČ	KAKO SE PREDSTAVLJA
ne	Slika ispred koje stoji <i>ne</i> je prekrižena
na	Jedna slika je na drugoj, između njih je linija
pod	Jedna slika je pod drugom, između njih je linija
u	Slovo se nalazi u drugom slovu
kraj	Između dvije slike nema razmaka
pored	Između dvije slike nema razmaka

Nedostaci i moguća poboljšanja

Program REGE uspješno generiše rebus za bilo koji tekst. Da bi dobijeni rebusi bili što zanimljiviji koriste se enigmatski trikovi (*ne, na, pod, kraj, pored, o*). Međutim, svi trikovi koje koriste enigmatičari nisu u potpunosti implementirani, kao na primjer prijedlog *u*. U rebusima, prijedlog *u* se često predstavlja tako što se jedna slika crta u drugoj. Ovo se ne koristi u programu pošto se svaka slika razlikuje, odnosno za svaku sliku pojedinačno bi trebalo odrediti mjesto gdje bi se ucrtavala druga slika. Program takođe ne raspoznaje ni mjesto na kome bi bilo najbolje da se napiše slovo. Iz tog razloga, da ne bi došlo do preklapanja slike i slova, slova se pišu pored slika (sa desna ili lijeva), što nije najbolje rješenje. Ovaj problem bi se mogao riješiti tako što bi se pamtile karakteristike svake slike, ali ovakvo rješenje bi otežalo postupak dodavanja novih slika.

Nedostatak je i što program za najbolje rješenje smatra onu riječ koja se u najvećoj mjeri poklapa sa traženom. Neće uzeti u obzir mogućnost da je možda bolje staviti riječ sa manje slova jer će se ostala sadržati u sledećoj riječi. Samim tim rebusi koji će se ovako dobiti ne moraju uvijek predstavljati najbolje rješenje.

Primjer kukati



Prva slika je rebus koji ćemo dobiti kao rješenje za kukati, a druga predstavlja kako bi to trebalo izgledati da program obraća pažnju na slova poslije neke nađene riječi. Jedan od načina da se prevaziđe ovaj problem je da se generiše više rebusa. Odnosno da se svaka riječ koja se poklapa sa početkom tražene koristi kao najbolja riječ za generisanje jednog rebusa, a riječ koju program okarakteriše kao najbolju (nezavisno da li se ona poklapa sa traženom ili ne) za generisanje drugog rebusa. Na taj način bi se za upisani tekst moglo dobiti više rebusa i korisnik bi mogao izabrati najbolji.

Literatura

Stojanović, B. 1981. *Rarebusi i rebusoidi*. Bjelovar: Enigmatsko udruženje ČVOR

REGE – Rebus Generator

Rebuses are riddles in pictures. The procedure, which takes us to a new rebus isn't simple at all. It takes a lot of inspiration and hard work. Rebuticians use their inspiration, so the rebuses can't be created simple by order. A small number of them are published and most of them are republished.

Because the computer doesn't get bored while trying to find the best solution for some sentences (while human does), an idea of making a program that would make rebuses according to the given sentence appeared. And that would be an easy job to get some better good and interesting rebuses.

The program is written in Object Pascal 7.0 (Borland Delphi). User interface is in Serbian. It should be mentioned that it's much easier to create a rebus in Serbian language than, for instance, in English, because one shouldn't worry about how the word is pronounced or spelled (written). Rebuses are based on pronunciation, so in English, the word *eye* = the pronunciation of the letter *i*, in contrast to phonetics Serbian where the pronunciation of the word is the same as the spelling of the same word (in Serbian to each letter corresponds one, and only one, sound) so it's much easier for the computer to find out which picture can replace what word or part of the word.

The rules for solving these rebuses (which are created by a computer) are the same as the rules used for rebuses made by a man. Of course, there are some standard replacements for pictures. For example, words (mostly prepositions) like in, on, by the, near are presented with one picture in, on, by, near some other picture. Also letters and apostrophes are used.

In fact, for every sentence entered by user, this program will create a rebus, for which solution is the entered sentence. Rebuses can be saved, and reused in other applications for further processing or printing.

