

## PRAKTIKUM IZ PROGRAMIRANJA 2

### - domaći zadatak broj 3 -

Sastaviti program na programskom jeziku C kojim se vrši određena vrsta obrade nad znakovnim nizom (u daljem tekstu, string). Program treba da repetitivno učitava sve potrebne ulazne podatke, ispisuje ih po učitavanju, izvršava zahtevanu obradu, ispisuje sve dobijene rezultate, i ponavlja navedene korake sve dok korisnik za ulazni string ne unese „`dosta Brus Li`”.

Zavisno od rednog broja problema, sastaviti **jedan** od sledećih programa koji:

0. utvrđuje da li u okviru stringa postoji jedna ili više `dancing` sekvenci, a zatim sve bele znakove između dve reči zamenjuje jednim blanko znakom; `dancing` sekvenca znakova je ona kod koje je prvo slovo veliko ili malo, a zatim se naizmenično smenjuju mala i velika slova uz ignorisanje belih znakova (npr. `PrA kTi KuM`);
1. svaki string koji predstavlja `for` petlju napisanu na C-u menja `for` petljom na jeziku Pascal; npr. `for(i=100; i>=a-b; i--)` se menja sa `for i:=100 downto a-b do`; pretpostaviti da svaki string koji počinje ključnom reči `for` predstavlja korektnu brojačku `for` naredbu na jeziku C; delovi naredbe mogu biti razdvojeni sa više od jednog blanko znaka;
2. svaki string koji predstavlja `for` petlju napisanu na C-u menja `while` petljom na jeziku Pascal; npr. `for(i=100; i>=a-b; i--)` `{...}` se menja sa `i:=100; while i>=a-b do begin ... i:=i-1; end`; pretpostaviti da svaki string koji počinje ključnom reči `for` predstavlja korektnu brojačku `for` naredbu na jeziku C; delovi naredbe mogu biti razdvojeni sa više od jednog blanko znaka;
3. iz stringa uklanja sve one reči koje se pojavljuju više od jedanput, a na mesto prvog pojavljivanja takve reči upisuje ukupan broj pojava u stringu;
4. vrši konverziju svih brojeve u stringu za kojima neposredno slede oznake za dužinu u imperijalnom mernom sistemu (`ft` za stope, `yd` za jarde i `ml` za milje) u metrički merni sistem; koristiti sledeće odnose:  $1\text{ m} = 3\text{ ft}$ ,  $1\text{ yd} = 0.914\text{ m}$ ,  $1\text{ ml} = 1650\text{ m}$ ;
5. vrši zamenu svih pojavljivanja cifara u stringu slovnim nazivima tih cifara; ukoliko postoji više cifara u nizu, razdvajati ih crticom prilikom prepisivanja;
6. vrši konverziju svih brojeve u stringu zadatih heksadecimalnom brojnem sistemu u dekadni brojni sistem; smatrati da heksadecimalni brojevi sadrže cifre 0-9 i slova `A-F` ili `a-f`, a da heksadecimalni broj uvek počinje prefiksom `0x` (npr. `0x9C13F`);
7. menja u stringu sve e-mail adrese oblika `identifikator@sajt.domen` u oblik `identifikator [at] sajt [dot] domen`; korektna adresa može sadržati veći broj znakova `.` a samo jedan znak `@`; pritom dva znaka `.` se ne mogu nalaziti jedan do drugog;

Dimenzije stringa nisu poznate unapred. Memorija predviđena za string mora biti **dinamički alocirana**. Količina memorije koja je odvojena za string mora biti takva da iskorišćenje bude potpuno (ni bajt više, ni bajt manje). Realokaciju vršiti uvek kada je potrebno, i prilikom unosa stringa i prilikom obrade. U toku rada, prilikom svake alokacije ili realokacije dinamičke memorije, proveravati uspešnost poziva `alloc` funkcije. U slučaju neuspešne dodele dinamičke memorije, u glavnom programu ispisati poruku o grešci i prekinuti izvršavanje. Nakon svakog ciklusa programa (unos, ispis unetih podataka, obrada, ispis dobijenih rezultata) dealocirati svu dinamičku memoriju. Smatrati da korisnik string zadaje u jednom redu teksta putem standardnog ulaza (u string ulaze svi znakovi do `\n`, ne uključujući i `\n`). **Obradu treba vršiti nad kopijom stringa, a pri ispisu rezultata na izlaz ispisati i neizmenjeni original i izmenjenu kopiju.** Pre odbrane, priložiti na listu papira makar tri suštinski različita test primera i očekivane rezultate.

**Važno: po uspešnom rešavanju osnovnog zadatka, modifikovati osnovni program tako zapamti svaki uneti red teksta i rezultat njegove obrade. Za smeštanje zapamćenih stringova koristiti dinamički niz stringova. Prilikom napuštanja programa, ispisati sve parove ulaznih stringova i rezultata njihove obrade.**

**Napomene:**

1. Odbrana trećeg domaćeg zadatka je u četvrtak, 11.05.2017. po rasporedu dostupnom na sajtu predmeta.
2. Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa):  **$i = (R + G) \bmod 8$**
3. Kao rešenje domaćeg zadatka potrebno je na odbrani pokazati sledeće datoteke:
  - **dz3.c**, koja sadrži izvorni tekst osnovnog programa na programskom jeziku C;
  - **dz3\_mod.c**, koja sadrži izvorni tekst izmenjenog programa na programskom jeziku C;

28.04.2017. godine

*sa predmeta*