|  |  |
| --- | --- |
| Univerzitet u Sarajevu  Elektrotehnički fakultet Sarajevo  Osnove računarstva | Rok za slanje zadaće:  *6. 1. 2016. (srijeda)* |

Zadaća 4

*Zadaća nosi 2 boda, a sastoji se od 4 zadatka. Svaki zadatak nosi po 0,5 bodova. Zadaci će biti bodovani proporcionalno broju uspješnih testova. Oblasti zadaće su: pokazivačka aritmetika te stringovi, uključujući napredne zadatke sa stringovima (P14-P17).*

*U ovoj zadaći je potrebno pridržavati se i pravila za pisanje čitljivog koda (P13), u suprotnom studenti mogu gubiti bodove.*

*Važna napomena: Ovaj dokument će možda biti mijenjan radi popravljanja grešaka. Koristite najnoviju verziju! Na dnu dokumenta nalaziće se dnevnik izmjena.*

**1. (0,5 bodova)** Potrebno je napisati funkciju pod imenom transformisi koja prima niz cijelih brojeva i obavlja dva zadatka pri čemu ako tačno uradite samo jedan od ta dva zadatka trebate dobiti najmanje 0,2 boda:

A) Ako se u nizu ponavlja isti broj ali sa različitim predznakom, i to najprije pozitivan a zatim negativan, potrebno je iz niza izbaciti sve članove niza između ta dva člana uključujući i ta dva.

Ovo treba uraditi isključivo ako je prvi broj (slijeva) pozitivan a drugi negativan, u obrnutom slučaju ne treba raditi izbacivanje.

Ako se isti članovi niza ponavljaju, funkcija treba pronaći prvi pozitivan broj (slijeva) i odgovarajući prvi negativan broj (slijeva) i izvršiti izbacivanje, pa ako nakon tog izbacivanja u nizu ostane još takvih parova izbacivanje se ponavlja itd. sve dok se više nema šta izbaciti.

Primjer: Neka je primljen niz:

1 5 2 5 3 -5 4 -5 5 -5

Najprije će biti uočen par 5 i -5 označen crvenom pozadinom jer je svaki od njih prvi takav član niza gledano slijeva, pa će biti izbačena ta dva člana i sve između njih (označeno žutom bojom), pa će zatim biti prepoznat par 5 i -5 označenih plavom bojom i izbačen (između njih se ne nalazi ništa) tako da će na kraju niz glasiti:

1 4 -5

B) Ako su u nizu dva susjedna člana takva da je drugi (slijeva) kvadrat prvog (npr. 4 i 16), između njih treba u niz ubaciti sve parne brojeve koji se nalaze na intervalu između prvog i drugog broja (ne uključujući ta dva). Ovo ubacivanje se treba obaviti nakon izbacivanja iz A). Primjer: dat je niz:

1 5 25 -1 2 3 9 8

Nakon poziva funkcije niz treba glasiti:

2 3 4 6 8 9 8

jer je 9 = 32, pa su između njih ubačeni brojevi 4, 6 i 8, a brojevi između 1 i -1 su izbačeni.

Funkcija treba po završetku vratiti broj članova niza nakon transformacije. Potrebno je napraviti i main program koji omogućuje da se testiraju obje funkcionalnosti tako što se najprije omogućuje korisniku da unese niz, zatim se poziva napravljena funkcija transformisi, te se na kraju rezultat ispisuje na ekranu.

**2. (0,5 bodova)** Napisati funkciju sa prototipom:

char\* centriraj(char\* str, int n)

koja centrira proslijeđeni string unutar širine n. Ukoliko je dužina stringa veća od n, funkcija ne

treba uraditi ništa. Primjer korištenja:

char n1[100] = "Bosna", n2[100] = "Hercegovina";

char\* p1 = centriraj(n1, 25);

char\* p2 = centriraj(n2, 25);

printf ("'%s'\n'%s'", p1, p2);

Kôd iznad treba da ispiše

'          Bosna          '

'       Hercegovina       '

Ukoliko centriranje nije moguće izvršiti ravnomjerno, sa desne strane se treba nalaziti razmak više.

U ovom zadatku nije dozvoljeno koristiti funkciju sprintf iz biblioteke string.h.

**3. (0,5 bodova)** Napisati funkciju sa prototipom:

char\* pretvori\_u\_bin(char\* str)

Koja pronalazi sve brojeve u stringu i mijenja ih njihovom binarnom reprezentacijom (koja počinje sa „0b“). Pretpostaviti da brojevi u stringu nisu veći od 1023 i da su pozitivni i cijeli. Ukoliko postoji realan broj (npr. „2.7“) treba ga tretirati kao dva cijela broja (tj. zamijeniti sa „0b10.0b111“). Riječi u stringu su razdvojene znakom razmak, znakom tačka ".", te znakom zarez "," (moguće više uzastopnih ponavljanja, npr. “broj    123,,, je ... 123”). Također, razmatrati samo brojeve koji su posebne riječi (tj. tekst „IM1“ ne treba zamijeniti sa „IM0b1“) ali tekst „IM2.7“ treba zamijeniti sa „IM2.0b111“ jer je i tačka separator za riječi. Izvorni string str će imati dovoljno mjesta da prihvati sva proširenja ukoliko do njih dođe.

Primjer:

char text[1000] = "Na OR-u ima 13 tutora od cega je 11 demonstratora. Ovo je 4. zadaca.";

pretvori\_u\_bin(text);

printf ("%s", text);

Primjer iznad treba da ispiše

Na OR-u ima 0b1101 tutora od cega je 0b1011 demonstratora. Ovo je 0b100. zadaca.

U ovom zadatku nije dozvoljeno koristiti funkcije sprintf iz biblioteke string.h niti itoa iz biblioteke stdlib.h.

**4. (0,5 bodova)** Potrebno je napraviti funkciju izracunaj koja obavlja računske operacije specificirane primljenim stringom i vraća njihov rezultat kao vrijednost tipa int. String se sastoji isključivo od sljedećih znakova: cifre 0-9, znakovi plus "+", minus "-", zvjezdica "\*", kosa crta "/" i strelica prema gore odnosno "kapa" "^".

Značenje stringa je sljedeće: cifre čine pozitivne cijele brojeve, a znakovi računske operacije i to:

^ stepenovanje

\* množenje

/ dijeljenje

+ sabiranje

- oduzimanje ili unarni minus (negacija)

Svi operatori su binarni, osim unarnog minusa, i imaju značenje kao što je uobičajeno u matematici. Najviši prioritet ima unarni minus, zatim stepenovanje, treći prioritet imaju množenje i dijeljenje, a četvrti prioritet sabiranje i oduzimanje. Kod operatora koji imaju isti prioritet evaluacija se vrši s lijeva na desno. Unarni minus se ne može više puta primijeniti na isti operand, tako da se string "--2" treba smatrati za neispravan. String "3--2" ima isto značenje kao "3+2" (jedan od dva minusa je unarni minus primijenjen na broj 2 a drugi je operacija oduzimanja).

Drugi parametar funkcije je pokazivač tipa int koji omogućuje da se sazna da li je primljeni string u ispravnom formatu, pa ako je primljeni string na bilo koji način neispravan u ono na šta pokazuje pokazivač treba upisati nulu, a u suprotnom jedinicu. Primjer upotrebe:

int ok;  
 int rez = izracunaj("2+2", &ok);

if (ok)

printf("rezultat je: %d\n", rez);  
 else

printf("greska\n");

Pretpostavite da se u stringu neće nalaziti više od 100 brojeva (operanada).

Potrebno je napraviti i kraći main program koji omogućuje da se testira da li funkcija izracunaj radi ispravno.

*Izmjene:*

* *23. 12. 2016 11:00 - u 4. zadatku dodato pojašnjenje za višestruku primjenu unarnog minusa.*
* *26. 12. 2016 20:10 - u zadacima 2. i 3. dodate zabranjene funkcije koje bi zadatke učinile trivijalnim. U četvrtom zadatku dodato još jedno pojašnjenje upotrebe unarnog minusa. U prvom zadatku dodata riječ "tačno" ("...ako tačno uradite samo jedan od ta dva…").*
* *29. 12. 2016 9:25 - u 4. zadatku dodata pretpostavka broja operanada.*