

Zadatak 1. (8 bodova) (2010./2011., ZI)

a) Napišite funkciju čiji je prototip

int izbaciVisestruke(char *niz);

Funkcija treba izbaciti sva višestruka ponavljanja nekog znaka u nizu i ostaviti samo zadnje pojavljivanje znaka. Funkcija treba vratiti broj izbacenih znakova. Primjer: niz "aaBzaZBBaCBZzZ" funkcija mijenja u niz "aCBzZ" i vraća broj 9.

b) Napišite glavni program u kojem će se pomoću funkcije **izbaciVisestruke** iz niza "Udovica baci vodu" izbaciti višestruki znakovi, te tako modificiran niz ispisati na zaslon.

Zadatak 2. (8 bodova) (2009./2010. ZI)

a) Napišite funkciju insert koja **bez korištenja pomoćnog polja** u zadani niz niz1 **iza znaka** s indeksom člana niza pozicija ubacuje niz niz2. Funkcija vraća pokazivač na niz1. Prototip funkcije je:

char *insert (char *niz1, char *niz2, int pozicija);

Primjer: za učitane nizove niz1 ABCDE, niz2 FG i poziciju 2 funkcija insert mijenja niz1 u ABCFGDE.

b) Napišite glavni program koji će s tipkovnice učitati nizove niz1 i niz2 i poziciju (indeks člana niza) **iza koje** niz2 treba ubaciti u niz1. Pozvati funkciju insert i na zaslon ispisati niz niz1.

Napomena: nije potrebno obavljati provjere duljina ulaznih nizova te vrijednost pozicije - pretpostavlja se da će svi uneseni podaci biti takvi da će se funkcija moći ispravno obaviti.

Zadatak 3. (8 bodova)(2008./2009. ZI)

Napisati funkciju čiji je prototip: void zamijeni (char *nizStari, char *nizNovi);

koja će niz znakova nizStari zadan u formatu "PREZIME, IME" prepisati u niz nizNovi, i to tako da niz nizNovi bude formata "IME PREZIME". Nije dozvoljeno mijenjanje niza nizStari.

Napisati glavni program u kojem se učitava niz znakova u formatu "PREZIME, IME" (pretpostavite da će niz biti unesen u tom formatu) te ispisuje u formatu "IME PREZIME".

Primjer: Za učitani niz "ANIC, ANA" potrebno je ispisati "ANA ANIC".

Zadatak 4. (8 bodova)(2007./2008. PZI)

void prepisi(char *prvi, char *drugi, char *treći).

Funkcija treba znakovni niz treći napuniti znakovima koji su zajednički nizovima prvi i drugi.

Ako se neki zajednički znak pojavljuje više puta treba ga prepisati samo jednom. Znakovi trebaju biti prepisani prema redoslijedu pojavljivanja u nizu drugi.

Primjer: Ako se prvi niz sastoji od znakova **nizPrvi**, a drugi od znakova **DrugiNiz** funkcija u treći niz treba prepisati znakove **riz**.

Napomena: Treba razlikovati mala i velika slova.

Riješenja:

Zadatak 1 (Bez uporabe string.h)

```
#include <stdio.h>
#include <stdlib.h>
```

```
int izbaciVisestruke(char *niz);
```

```
int main()
{
    char niz[51] = "Udovica baci vodu";
    int br;
    printf("%s\n", niz);
    br = izbaciVisestruke(niz);
    printf("%s\n", niz);
    printf("%d", br);

    return 0;
}
```

```
int izbaciVisestruke(char *niz)
```

```
{
    int count = 0, strln, i, j, k;

    for(i = 0, strln = 0;;i++)
        if(*(niz + i) != '\0') strln++; // Vraća nam duljinu stringa, ne uključujući null char, čije nalaženje
        else break;                     // prekida brojanje i ne povisuje brojač

    for(i = 0; i < strln; i++)
    {
        for(j = i + 1; j < strln; j++) //Za svaki član niza provjerava daljnje članove niza, počevši od
        {                             // onoga iza sebe (da ne prebriše sam sebe!)
            if(*(niz + i) == *(niz + j)) //Ako nade isti
            {
                for(k = i; k < strln; k++) //Svi iza se kopiraju na mjesto prije, od člana i+1 do kraja niza
                    *(niz + k) = *(niz + k + 1);
                count++; //Brojač se povisuje po opisu funkcije
                --i; --strln; //Niz je sad kraći za 1 član. I se smanjuje, jer je sad na mjestu i novi član, koji
                break; // bi bio preskocen kad se i poveća. Break izlazi iz 2. for petlje
            }
        }
    }
    return count;
}
```

Zadatak 2:

```
#include <stdio.h>
#include <ctype.h>
#include <string.h>
#include <stdlib.h>
#define MAXSTRLEN 51
```

```
char *insert (char *niz1, char *niz2, int pozicija);
```

```
int main()
{
    char niz1[MAXSTRLEN], niz2[MAXSTRLEN];
    int poz;
    printf("Upisite prvi string:\n");
    gets(niz1);
    printf("Upisite string koji zelite umetnuti:\n");
    gets(niz2);
    printf("Upisite clan stringa 1 iza kojeg zelite umetnuti string 2:\n");
    scanf("%d", &poz);
    insert(niz1, niz2, poz);
    puts(niz1);

    return 0;
}
```

```
char *insert (char *niz1, char *niz2, int pozicija)
{
    strcat(niz2, niz1 + pozicija); //Preslikava dio niza1 nakon mjesta gdje ce se umetnuti niz2 na
    strcpy((niz1 + pozicija), niz2); //kraj samog niza 2. Niz2 se kopira preko niza iza mjesta pozicija,
    return niz1; //buduci da je niz2 + 2. dio niz1 dulji od 2. dio niz1, pa on bude prebrisan kopiranjem
}
```

Zadatak 3:

```
#include <stdio.h>
#include <ctype.h>
#include <string.h>
#include <stdlib.h>
#define MAXSTRLEN 51

void zamijeni(char *nizStari, char *nizNovi);

int main()
{
    char niz1[MAXSTRLEN], niz2[MAXSTRLEN];
    gets(niz1);
    zamijeni(niz1, niz2);
    puts(niz2);

    return 0;
}

void zamijeni(char *nizStari, char *nizNovi)
{
    int prezime_len = 0, i, ime_len = 0; //Inicijaliza duljine na 0

    for(i = 0;;i++)
        if(nizStari[i] != ',') prezime_len++; //Broji clanove do zareza, dajuci nam duljinu prezimena
        else break;

    ime_len = strlen(nizStari + prezime_len); //Cita duljinu imena, koje pocinje na pocetku niza +
                                                //duljina prezimena
    strncpy(nizNovi, (nizStari + prezime_len + 2), ime_len); //Prvo kopira ime, kad znamo gdje
    strcat(nizNovi, " "); //počinje i imamo njegovu točnu duljinu. Zatim na to dodaje razmak i onda
    strncat(nizNovi, nizStari, prezime_len); // i prezime

    return;
}
```

Zadatak 4:

```
#include <stdio.h>
#include <ctype.h>
#include <string.h>
#include <stdlib.h>
#define MAXSTRLEN 51

void prepisi(char *prvi, char *drugi, char *treci);

int main()
{
    char niz1[MAXSTRLEN], niz2[MAXSTRLEN], niz3[MAXSTRLEN];
    gets(niz1);
    gets(niz2);
    prepisi(niz1, niz2, niz3);
    puts(niz3);

    return 0;
}

void prepisi(char *prvi, char *drugi, char *treci)
{
    int i;
    for(i = 0; i < strlen(drugi); i++) //Prolazi kroz drugo polje, buduci da je zadano da se
    { //znakovi kopiraju redom kojim se pojavljuju u drugom polju, od 0 do strlen(drugi)
        if((strchr(prvi, *(drugi + i)) != NULL) && (strchr(treci, *(drugi + i)) == NULL))
        { //Tj. Ako postoji taj char u prvom nizu i ako vec nije u trecem (tj. Nije duplikat u nizu 2)
            treci[j] = drugi[i]; //Ako je to takav lik kopira se u niz 3
        }
    }
    strcat(treci, ""); //Dodaj prazan niz, tj. nul character, na sam kraj niza
}
```