

**Napomene:**

- Savjetuje se navedene zadatke riješiti ubrzo nakon predavanja
- Savjetuje se ne gledati rješenja prije nego se pokuša samostalno riješiti zadatke

## 22. vježbe uz predavanja

1. Na svom računalu testirajte sve primjere s predavanja.
2. Napisati program koji pomoću funkcije `getchar` s tipkovnice učitava znakove u niz `s1` dok god ne bude učitani znak 'X' (znak 'X' se ne dodaje u niz `s1`). Zatim u niz `s2` učitava znakove dok se ne pojavi znak 'Y' (znak 'Y' se ne dodaje u niz `s2`). Osigurati da nizovi `s1` i `s2` budu ispravno terminirani (tj. na kraju niza se nalazi znak '\0'). Nakon učitavanja, pomoću funkcije `putchar` ispisati nizove `s1` i `s2` na zaslon, svaki u svoj redak. Nizovi `s1` i `s2` sigurno nisu dulji od 80 znakova.

Primjer: ako se preko tipkovnice unese:

```
Ovo je
neki tekst.XA ovo
je nastavakY0statak se ne učitava
```

kao rezultat treba ispisati:

```
Ovo je
neki tekst.
A ovo
je nastavak
```

3. Napisati program kojim će se u dvodimenzijско znakovno polje učitati tekst s tipkovnice. Svaki redak teksta treba učitati u zasebni redak polja. Učitavanje redaka teksta treba prekinuti kad se učitava prazan redak ili se dođe do granice od najviše 10 dozvoljenih redaka. Može se računati da niti jedan redak teksta neće biti dulji od 80 znakova. Za čitanje teksta koristiti funkciju `gets` iz `<stdio.h>`. Na kraju, pomoću funkcije `printf`, na zaslon ispisati sve učitane retke teksta, svaki redak teksta u svom retku na zaslonu.

## Rješenja

### Rješenje 2. zadatka

```
#include <stdio.h>

int main(void) {
    char s1[80+1], s2[80+1], c;
    int i;

    i = 0;
    while((c = getchar()) != 'X')
        s1[i++] = c;
    s1[i] = '\0';

    i = 0;
    while((c = getchar()) != 'Y')
        s2[i++] = c;
    s2[i] = '\0';

    i = 0;
    while (s1[i] != '\0')
        putchar(s1[i++]);

    printf("\n");

    i = 0;
    while (s2[i] != '\0')
        putchar(s2[i++]);

    return 0;
}
```

### Rješenje 3. zadatka

```
#include <stdio.h>

#define MAXRED 10
#define MAXSTUP 80

int main (void) {
    char polje[MAXRED][MAXSTUP+1];
    int i, ucitanoRedaka = 0;

    while (ucitanoRedaka < 10) {
        /* gets kao argument koristi adresu prvog clana odgovarajućeg
           retka 2D polja */
        gets(&polje[ucitanoRedaka][0]);
        if (polje[ucitanoRedaka][0] == '\0')
            /* ucitan je prazan redak, prekini ucitavanje */
            break;
        ++ucitanoRedaka;
    }
    for (i = 0; i < ucitanoRedaka; ++i)
        printf("%s\n", &polje[i][0]);
    return 0;
}
```