

INFORMATIČKI KLUB
FUTURA

```
#include<stdio.h> C  
int main()  
{ RADIONICE PROGRAMIRANJA  
  printf("Hello World!");  
  ZA SREDNJE ŠKOLE  
  return 0;  
}
```

RADIONICE PROGRAMIRANJA ZA SREDNJE ŠKOLE - 3. RADIONICA

Krunoslav Žubrinić, Informatički klub FUTURA
Dubrovnik, 29. studenog 2014.



Creative Commons



slobodno smijete:

- **dijeliti** — umnožavati, distribuirati i javnosti priopćavati djelo
- **remiksirati** — prerađivati djelo



pod slijedećim uvjetima:

- **imenovanje**. Morate priznati i označiti autorstvo djela na način kako je specificirao autor ili davatelj licence (ali ne način koji bi sugerirao da Vi ili Vaše korištenje njegova djela imate njegovu izravnu podršku).
- **nekomercijalno**. Ovo djelo ne smijete koristiti u komercijalne svrhe.
- **dijeli pod istim uvjetima**. Ako ovo djelo izmijenite, preoblikujete ili stvarate koristeći ga, preradu možete distribuirati samo pod licencom koja je ista ili slična ovoj.



U slučaju daljnjeg korištenja ili distribuiranja morate drugima jasno dati do znanja licencne uvjete ovog djela. Najbolji način da to učinite je linkom na ovu internetsku stranicu.

Od svakog od gornjih uvjeta moguće je odstupiti, ako dobijete dopuštenje nositelja autorskog prava.

Ništa u ovoj licenci ne narušava ili ograničava autorova moralna prava.

Tekst licence preuzet je s <http://creativecommons.org/>.

Sadržaj

- Ponavljanje tema s 2. radionice
- Rad sa znakovima i nizovima znakova



Polje

```
int a[7];
```

Naziv	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]	a[6]
Vrijednost	12	45	32	23	17	49	5

U memoriji su varijable polja smještene jedna iza druge, a pristupa im se pomoću indeksa

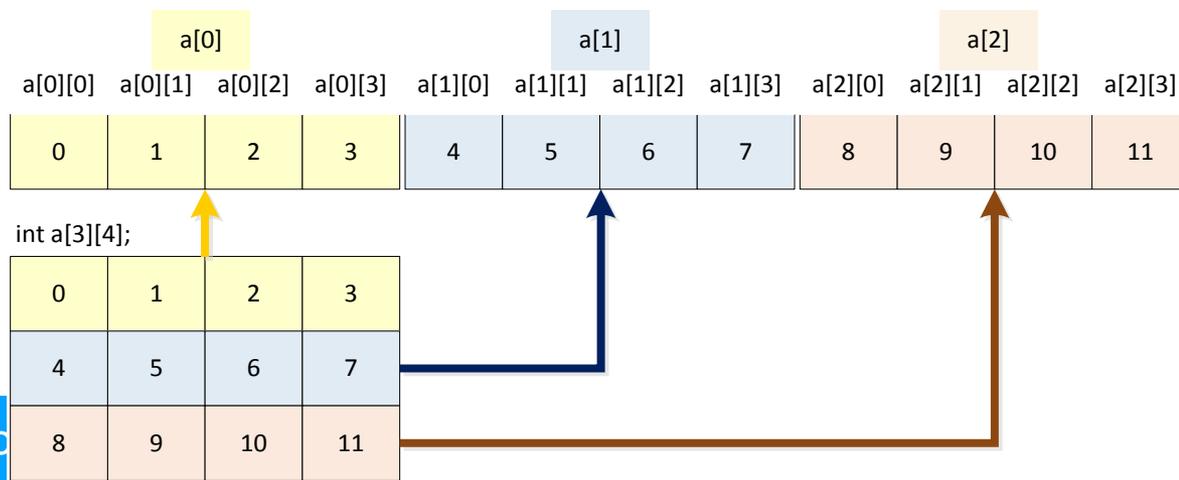
Polje može sadržavati bilo koji tip podataka

Varijabla	Sadržaj
int p[100]	Polje koje sadrži devedeset cijelih brojeva
float p[74]	Polje koje sadrži sedamdeset i četiri realna broja tipa float
double p[21]	Polje koje sadrži dvadeset i jedan realan broj tipa double
char p[30]	Polje koje sadrži trideset znakova

Polje može sadržavati čak i druga polja

int p[3][4]

Polje cijelih brojeva od 3 elementa, a svaki element je polje od 4 cijela broja



Rad s nizovima znakova



- U C-u se niz znakova bilježi kao polje podataka tipa **char**.
 - char se čita kao "**kar**" (od "**karakter**"), a NE "**čar**" (nije "**čaračter**" 😊)
- Polje završava posebnim znakom "završetka niza" - **\0**
- Primjerice ako je niz znakova „**Futura**”
 - varijabla je **polje duljine 7 znakova** (6 za tekst i 1 za oznaku kraja niza)
char niz[6+1]; odnosno **char niz[7];**
- Pristup jednom po jednom znaku niza obično se radi **while** petljom

char niz[7];

niz[0] niz[1] niz[2] niz[3] niz[4] niz[5] niz[6]

F	u	t	u	r	a	\0
---	---	---	---	---	---	----

```
while (niz[i] != '\0') {  
    printf("%c", niz[i]);  
    if (niz[i] == 'x')  
        break;  
    i++;  
}
```

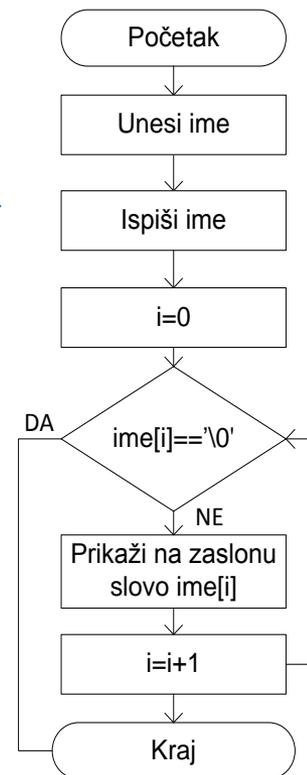
While petlja se izvodi **sve dok je uvjet u zagradama istinit**. Ako na početku taj uvjet nije istinit, petlja se neće izvesti niti jednom.

Iz petlje se može izaći u bilo kojem trenutku pozivom naredbe **break**.



Unos i ispis imena

Algoritam



Rezultat izvođenja



```
Unesite ime učenika/ce:Ivica
Uneseno ime je: Ivica
Uneseno ime je: Ivica
```



Ispis niza znakova na dva načina:

- Pomoću oznake **%s** (odjednom čitav niz)
- U petlji jedan po jedan znak pomoću oznake **%c**.



Anatomija programa

```
#include<stdio.h>
#include<stdlib.h>
int main(){
    char ime[20 + 1];
    int i = 0;
    printf("Unesite ime ucenika/ce:");
    scanf("%20s", ime);
    printf("Uneseno ime je: %s\n", ime);
    printf("Uneseno ime je: ");
    while (ime[i] != '\0'){
        printf("%c", ime[i]);
        i++;
    }
    printf("\n");
    system("pause");
    return 0;
}
```

Polje u koje se može spremiti 20 znakova

Unos niza znakova (maksimalno 20) **%20s**.
Naziv polja je **adresa prvog elementa** pa u ovom slučaju **ne treba adresni operator &** (kao što primjerice treba kod unosa cijelog broja).
ime je isto kao da smo napisali **&ime[0]**

Ispis kompletnog niza znakova na zaslon (oznaka **%s**)

Petlja prolazi kroz čitav niz – znak po znak dok ne dođe do kraja (znak='\0').

Ispis jednog znaka iz niza

Iz petlje se može izaći u bilo kojem trenutku pozivom naredbe **break!**

Unos i ispis imena

```
#include<stdio.h>
#include<stdlib.h>
int main(){
    char ime[20 + 1];
    int i = 0;
    printf("Unesite ime ucenika/ce:");
    scanf("%20s", ime);
    printf("Uneseno ime je: %s\n", ime);
    printf("Uneseno ime je: ");
    while (ime[i] != '\0'){
        printf("%c", ime[i]);
        i++;
    }
    printf("\n");
    system("pause");
    return 0;
}
```



Rad s nizovima znakova



- Unos niza kod deklaracije

```
char niz[] = "Radionica";
```

- Unos niza s prazninama sa standardnog ulaza (sve do „entera”)

```
scanf(" %[^\\n]", ime);
```

- Kopiranje **niza1** (stari niz) u **niz2** (novi niz)

```
#include<string.h>
```

```
strcpy (niz2, niz1);
```

- Lijepljenje **niza1** u **niz2** (novi niz)

```
#include<string.h>
```

```
strcat (niz2, niz1);
```

- Uspoređivanje dva niza **niza1** i **niz2**

```
#include<string.h>
```

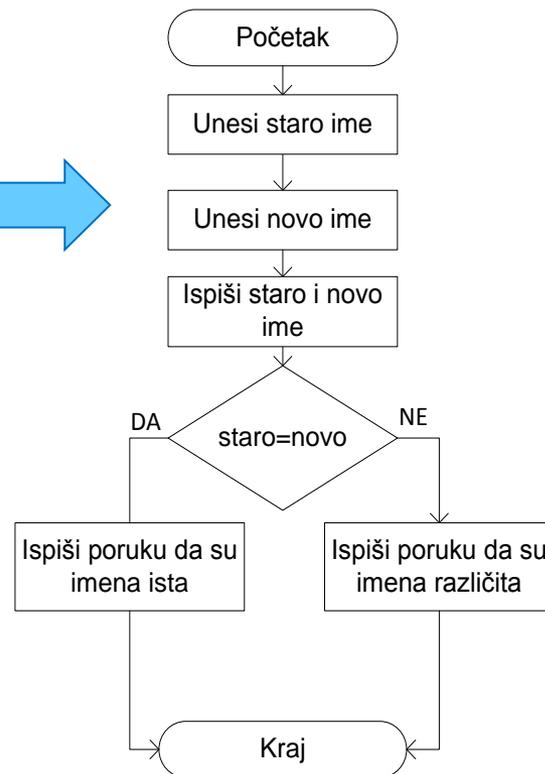
```
int strcmp (niz1, niz2);
```

- Ako su oba niza jednaka, funkcija vraća 0
- Ako je prvi niz manji od drugoga, funkcija vraća broj < 0
- Ako je prvi niz veći od drugoga, funkcija vraća broj > 0

Uspoređivanje imena



Algoritam



Rezultat izvođenja



```
Unesite ime i prezime ucenika/ce: Ana Anic  
Staro ime je Pero Peric, a uneseno: Ana Anic  
Imena su razlicita!
```



Anatomija programa

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
int main(){
    char imep[30 + 1];
    char staroime[] = "Pero Peric";
    printf("Unesite novo ime i prezime ucenika/ce:");
    scanf("%30[^\n]", imep);
    printf("Staro ime je %s, a uneseno:");
    if (strcmp(staroime, imep) == 0)
        printf("Staro i novo ime su jednaki!\n");
    else
        printf("Imena su razlicita!\n");
    system("pause");
    return 0;
}
```

Funkcije za rad s **nizovima znakova** opisane su u datoteci **string.h**

Polje u koje se može spremiti 30 znakova

Stvaranje polja znakova i pridruživanje početnog niza tom polju. Duljina polja je 11 (vidljiva slova + '\0')

Unos niza znakova duljine 30 (**uključujući praznine**) sve do pritiska na tipku ENTER (oznaka **%[^\n]**)

Ako su oba niza jednaka, funkcija vraća 0

Uspoređivanje imena

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
int main(){
    char imep[30 + 1];
    char staroime[] = "Pero Peric";
    printf("Unesite ime i prezime ucenika/ce:");
    scanf("%30[^\n]", imep);
    printf("Staro ime je %s, a uneseno: %s.\n", staroime, imep);
    if (strcmp(staroime, imep) == 0)
        printf("Staro i novo ime su jednaki!\n");
    else
        printf("Imena su različita!\n");
    system("pause");
    return 0;
}
```



Spajanje imena i prezimena



Algoritam



Rezultat izvođenja



```
Unesite ime ucenika/ce: Pero
Unesite prezime ucenika/ce: Peric
Ime i prezime je Pero Peric.
```



Anatomija programa

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
int main(){
    char ime[20 + 1], prezime[20 + 1], imep[41 + 1];
    printf("Unesite ime ucenika/ce:");
    scanf("%20[^\n]", ime);
    printf("Unesite prezime ucenika/ce:");
    scanf("%20[^\n]", prezime);
    strcpy(imep, ime);
    strcat(imep, " ");
    strcat(imep, prezime);
    printf("Ime i prezime je %s.\n", imep);
    system("pause");
    return 0;
}
```

Funkcije za rad s **nizovima znakova** opisane su u datoteci **string.h**

Voditi računa o veličini polja. Novo polje mora moći spremiti sadržaj svih polja koja se u njega spajaju.

[^\n] ne uzima u obzir znak ENTER (\n) pa se on prenosi na sljedeći scanf (preskače se unos).
Da se to izbjegne, **ispred znaka % stavite jednu prazninu.**

Polje imep je na početku prazno, pa se ne može odmah koristiti funkcija strcat. Inicijalno punjenje varijable imep ide funkcijom strcpy.

Nakon što je u varijabli imep nešto zapisano, spajanje se nastavlja funkcijom strcat.

Spajanje imena i prezimena

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
int main(){
    char ime[20 + 1], prezime[20 + 1], imep[41 + 1];
    printf("Unesite ime ucenika/ce:");
    scanf(" %20[^\n]", ime);
    printf("Unesite prezime ucenika/ce:");
    scanf(" %20[^\n]", prezime);
    strcpy(imep, ime);
    strcat(imep, " ");
    strcat(imep, prezime);
    printf("Ime i prezime je %s.\n", imep);
    system("pause");
    return 0;
}
```



1. zadatak: Ubbi Dubbi



Na nagradnoj igri osvojili ste putovanje na egzotičan tropski otok čiji stanovnici govore **Ubbi Dubbi** jezikom. Pravila Ubbi Dubbi jezika su takva da se riječima engleskog jezika ispred samoglasnika dodaju slova **ub**. Ako se u riječi samoglasnici u nizu ponavljaju više puta, slova **ub** se dodaju samo ispred prvog samoglasnika u tom nizu.

Na putovanju se želite što bolje sporazumijevati sa stanovnicima otoka, pa trebate pomoć. Napravite program koji će služiti za prevođenje pojedinačnih riječi **engleskog jezika u Ubbi Dubbi jezik**.

ULAZNI PODACI:

- Niz slova maksimalne duljine 20 znakova (mala slova a-z).

IZLAZNI PODACI:

- Niz slova preveden na Ubbi Dubbi jezik.

ULAZ

IZLAZ

speak

spubeak

beard

bubeard

ubbi

ububbubi

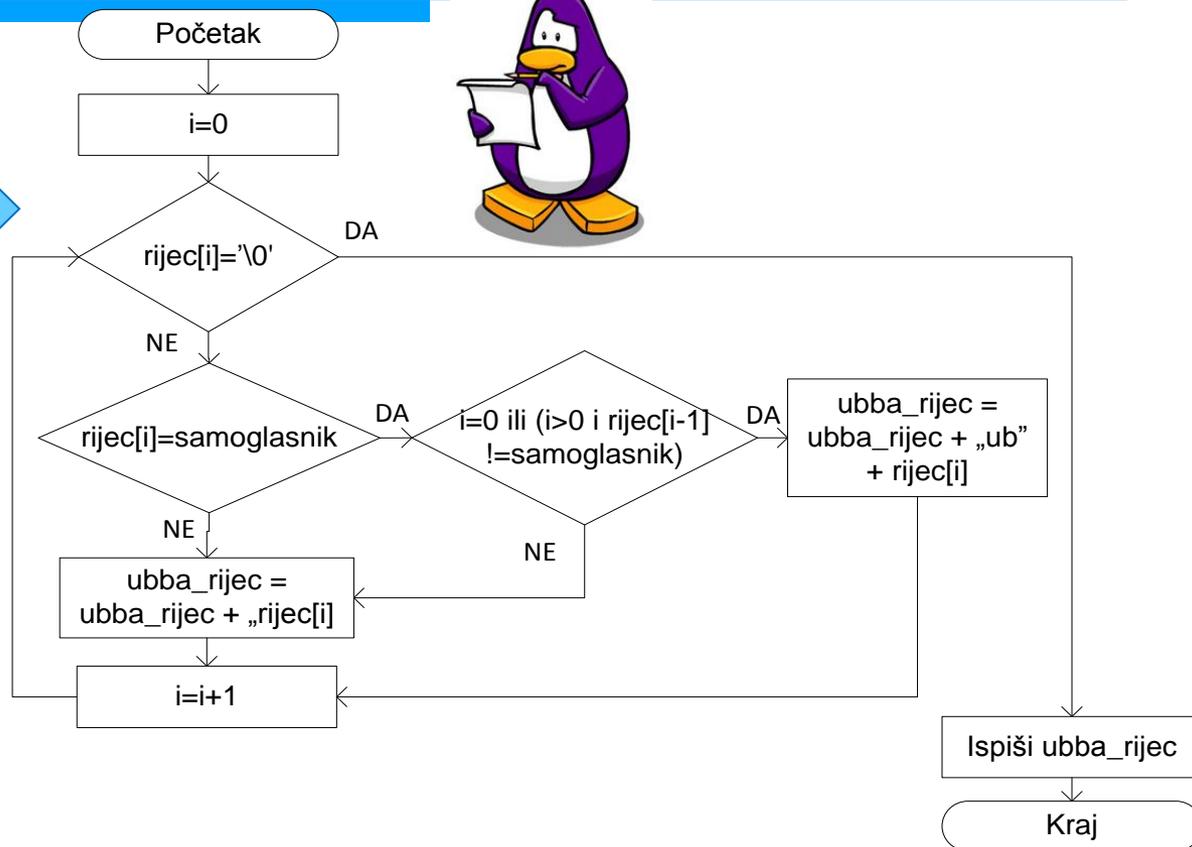
Vrijeme



Rješenje 1. zadatka



Algoritam



Rezultat
izvođenja



```
Engleska rijec: speak
Ubba rijec: spubeak
```

Rješenje 1. zadatka

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int i=0, j=0, nasao = 0;
    char ub[] = "ub", rijec[20+1], ubba_rijec[40+1];
    printf("Engleska rijec: ");
    scanf("%s", &rijec);
    while (rijec[i] != '\0'){
        if ((rijec[i] == 'a' || rijec[i] == 'e' || rijec[i] == 'i' || rijec[i] == 'o'
|| rijec[i] == 'u') && ((i == 0) || (rijec[i-1] != 'a' && rijec[i-1] != 'e' &&
rijec[i-1] != 'i' && rijec[i-1] != 'o' && rijec[i-1] != 'u'))){
            ubba_rijec[j] = 'u';
            ubba_rijec[j+1] = 'b';
            j = j + 2;
        }
        ubba_rijec[j] = rijec[i];
        j++;
        i++;
    }
    ubba_rijec[j] = '\0';
    printf("Ubba rijec: %s\n", ubba_rijec);
    system("pause");
    return 0;
}
```



Programski kod



2-tjedni izazov



Dovršite prevodioc iz četvrtog zadatka tako da može prevoditi rečenice u oba smjera: s engleskog na Ubbi Dubbi, ali i s Ubbi Dubbi jezika na engleski jezik.

ULAZNI PODACI:

- Slovo **e** ili **u** koje određuje smjer prevođenja

e = engleski ➡ Ubbi Dubbi,

u = Ubbi Dubbi ➡ engleski

- Niz slova (mala slova a-z i praznine) maksimalne duljine 200 znakova.

IZLAZNI PODACI:

- Niz slova preveden na Ubbi Dubbi ili engleski jezik, ovisno o prvom ulaznom podatku.

ULAZ

e
hello to all

IZLAZ

hubellubo tubo uball

ULAZ

u
ubi hubavube twubo
hubands

IZLAZ

i have two hands

ULAZ

e
do you speak english

IZLAZ

dubo yubou spubeak
ubenglubish

ULAZ

u
ubit ubis my
lubangubuagube

IZLAZ

it is my language