

01001111 01110011 01101110 01101111
01110110 01101001 00100000 01110000
01110010 01101111 01100111 01110010
01100001 01101101 01101001 01110010
01100001 01101110 01101010 01100001



OSNOVI PROGRAMIRANJA

VEŽBE 11

- $\text{LENGTH}(\text{Str})$ – određuje dužinu datog stringa **Str**

Pr. $a := \text{'Kragujevac'}$

$i := \text{length}(a) \rightarrow i = 10$

- $\text{CONCAT}(\text{St1}, \text{St2}, \dots, (\text{Stn}))$ – sabira stringove **St1, St2, ..., Stn**

Pr. $a := \text{'Novi'}$

$b := \text{'Sad'}$

$c := \text{concat}(a, b) \rightarrow c = \text{'Novi Sad'}$

- $\text{POS}(\text{Str1}, \text{Str2})$ – određuje vrednost pozicije od koje se string **Str1** nalazi u stringu **Str2**

Pr. $i := \text{pos}(\text{'ra'}, \text{'Kragujevac'}) \rightarrow i = 2$

- $\text{COPY}(\text{St}, \text{Poz}, \text{Br})$ – kopira string **St** od pozicije **Poz** i kopira **Br** simbola

Pr. $a := \text{'Beograd'}$

$b := \text{copy}(a, 4, 2) \rightarrow b = \text{'gr'}$



- `DELETE (St, Poz, Br)` – briše string **St** od pozicije **Poz** i briše **Br** simbola.

Pr. `a := 'Kragujevac'`

`delete(a, 7, 4) → a = 'Kraguj'`

- `INSERT (StUb, StObj, Br)` – u string **StObj** ubacuje u string **StUb** od pozicije **Br**

Pr. `a := 'Beod'`

`b := 'gra'`

`insert(b, a, 4) → a = 'Beograd'`

- `STR (NumVr, Str)` – Pretvara broj **NumVr** u string **Str**

Pr. `str(567, s) → s = '567'`

- `VAL (Str, NumVr, Kod)` – pretvara string **Str** u broj **NumVr**, ako je string preved bez greške **Kod** dobija vrednost 0, u suprotnom vrednost pozicije na kojoj je greška nastala

Pr. `val('345', i, k) → i = 345`



- Napisati program koji uneti string u formatu
Prezime*Broj*Ime
prevodi u format
Ime Prezime Broj

```
Program OP11_1;  
Var s,p,i,b:string;  
    n:integer;  
Begin  
    readln(s);  
    n:=pos('*',s);  
    p:=copy(s,1,n-1);  
    delete(s,1,n);  
    n:=pos('*',s);  
    b:=copy(s,1,n-1);  
    delete(s,1,n);  
    s:=concat(s,' ',p,' ',b);  
    writeln(s)  
End.
```



- Napisati f-ju koji za unetu reč određuje da li je palindrom ili ne. Za niz unetih reči štampati palindrome među njima.

```
program OP11_2;
var i,n,:integer;
    a:array[1..50]of string;

function palindrom(s:string):boolean;
var i,n:integer;
begin
    n:=length(s);
    i:=1;
    while (s[i]=s[n-i+1]) and (i<=n div 2) do i:=i+1;
    if i>n div 2 then palindrom:=true
    else palindrom:=false;
end;

begin
    readln(n);
    for i:=1 to n do readln(a[i]);
    for i:=1 to n do
        if (palindrom(a[i])) then writeln(a[i]);
    End.
```



1. Napisati potprogram **UnosNizaS** koji učitava niz stringova.
2. Napisati potprogram **IspisNizaS** koji ispisuje niz stringova ispod drugog.
3. Napisati potprogram **SamoSlova** koja ispituje da li je zadati string sastavljen samo od slova engleskog alfabeta.
4. Napisati potprogram **DvaUzastopna** koji ispituje da li se u zadatom stringu nalaze dva ista znaka jedan pored drugog.
5. Napisati potprogram **VelikoSlovo** koji ispituje da li se u zadatom stringu nalazi neko veliko slovo a da to nije na početku stringa.
6. Za neki string kažemo da je **korektna reč** ako ima bar dva karaktera, sastoji se samo od slova engleskog alfabeta, nema dva ista slova jedno do drugog i počinje velikim slovom koje je ujedno i jedino veliko slovo u toj reči. Napisati potprogram **KorektnaRec** koji za zadatu reč ispituje da li je korektna ili ne.
7. U glavnom delu programa učitati niz stringova, kreirati niz stringova samo sa korektnim rečima, i potom ga ispisati.

```

program OP_11_3;
type tnizs = array[1..50] of string;
var i,j,n:integer;
    a,b:tnizs;
procedure UnosS(var a:tnizs; var n:integer);
var i:integer;
begin
    readln(n);
    for i:=1 to n do
        readln(a[i]);
    end;
procedure IspisS(a:tnizs; n:integer);
var i:integer;
begin
    for i:=1 to n do
        writeln(a[i]);
    end;
function SamoSlova(s:string):boolean;

```

```

var ind:boolean;
    i:integer;
begin
    ind:=true;
    for i:=1 to length(s) do
        if (s[i] < 'A') or ((s[i] > 'Z') and
            (s[i] < 'a')) or (s[i] > 'z') then
            ind:=false;
        SamoSlova:=ind;
    end;

```

```

begin
    UnosS(a,n);
    j:=0;
    for i := 1 to n do
        if(KorektnaRec(a[i])) then
            begin
                j:=j+1;
                b[j]:=a[i];
            end;
    IspisS(b,j);
end.

```

```

function DvaUzastopna(s:string):boolean;
var i:integer;
    ind:boolean;
begin
    ind:=false;
    for i:=1 to length(s)-1 do
        if(s[i] = s[i+1])then
            ind:=true;
        DvaUzastopna:=ind;
    end;
function VelikoSlovo(s:string):boolean;
var i:integer;
    ind:boolean;
begin
    ind:=false;
    for i:=2 to length(s) do
        if(s[i] >= 'A') and (s[i] <= 'Z') then
            ind:=true;
        VelikoSlovo:=ind;
    end;
function KorektnaRec(s:string):boolean;
begin
    if(SamoSlova(s)and
        (not DvaUzastopna(s))and
        (not VelikoSlovo(s))) then
        KorektnaRec:=true
    else
        KorektnaRec:=false;
    end;

```

- Napisati program koji iz tekstualne datoteke **TEKST1.TXT** čita tekst i određuje broj pojavljivanja svih malih slova engleske abecede u tekstu. U datoteku **TEKST2.TXT** ispisati sva slova koja se pojavljuju u datoteci **TEKST1.TXT** i broj njihovih pojavljivanja, tako što svaki red sadrži slovo i njegov broj pojavljivanja razdvojen pauzom.

```
program OP11_4;
var a:array['a'..'z'] of integer;
    i:integer;
    c:char;
    f:text;
begin
    for c:='a' to 'z' do a[c]:=0;
    assign(f,'tekst1.txt');
    reset(f);
    while not eof(f) do begin
        read(f,c);
        if (c>='a') and (c<='z') then a[c]:=a[c]+1;
    end;
    close(f);
    assign(f,'tekst2.txt');
    rewrite(f);
    for c:='a' to 'z' do
        if a[c]>0 then writeln(f,c,' ',a[c]);
    close(f)
end.
```

- Napisati program koji na ulazu učitava imena dve datoteke. U prvoj datoteci se nalazi tekst koji treba prepisati u drugu datoteku, ali na početku svakog reda dodati broj reda, prazninu, a zatim tekst iz ulazne datoteke. Na ekranu ispisati broj reda u kome ima najviše karaktera.



```
program OP11_5;
var f1,f2:text;
    s1,s2:string;

function Najduza(var fin,fout:text):integer;
var max,br,n:integer;
    s:string;
begin
    n:=0;
    br:=0;
    max:=0;
    while not eof(fin) do
    begin
        n:=n+1;
        readln(fin,s);
        if max<length(s) then
        begin
            max:=length(s);
            br:=n
        end;
        write(fout,n,' ');
        writeln(fout,s);
    end;
    Najduza:=br
end;
```

```
begin
    write('Ime ulazne datoteke:'); readln(s1);
    write('Ime izlazne datoteke:'); readln(s2);
    assign(f1,s1);
    reset(f1);
    assign(f2,s2);
    rewrite(f2);
    writeln(Najduza(f1,f2));
    close(f1);
    close(f2)
end.
```