

Programski jezik PASCAL

Korak po korak

1. dio

dipl.ing.el. Emir Zečić

Sarajevo, februar 2014.

Poglavlja

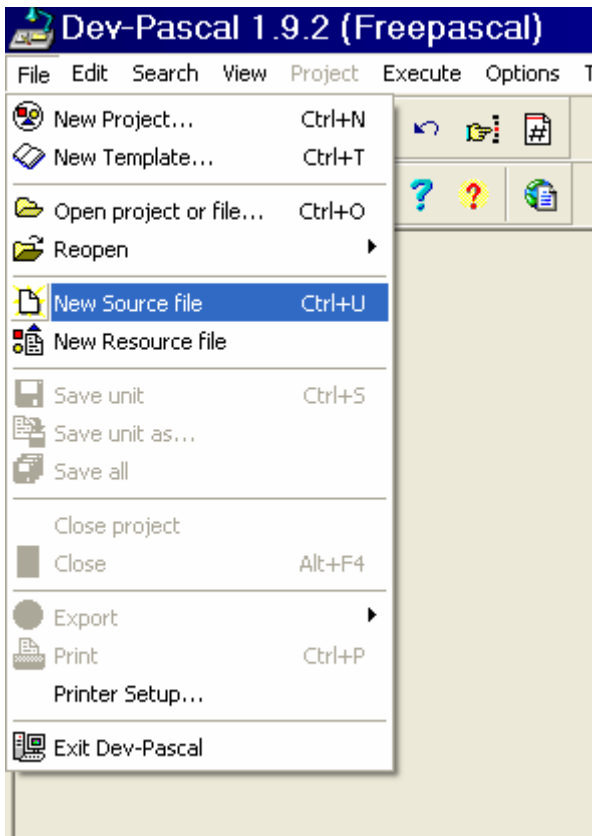
- Instalacija kompajlera i primjer „Hello World“ programa
- Varijable, writeln i readln naredbe
- Iskaz pridruživanja
- Standardne funkcije
- Jednostavni IF ... THEN uslov
- Blok na primjeru IF ... THEN
- AND i OR na primjeru IF ... THEN
- IF ... THEN ... ELSE
- IF grananje
- CASE naredba

POGLAVLJE 1: Instalacija kompajlera i primjer „Hello World“ programa

1) Da bi se napisao program u programskom jeziku Pascal potreban je kompajler. Kompajler se može preuzeti sa lokacije :

<http://www.bloodshed.net/dev/devpas192.exe>

2) Nakon instalacije i pokretanja programa izabrati opciju **File / New Source File**



3) U novootvorenom prozoru napisati kostur programa.

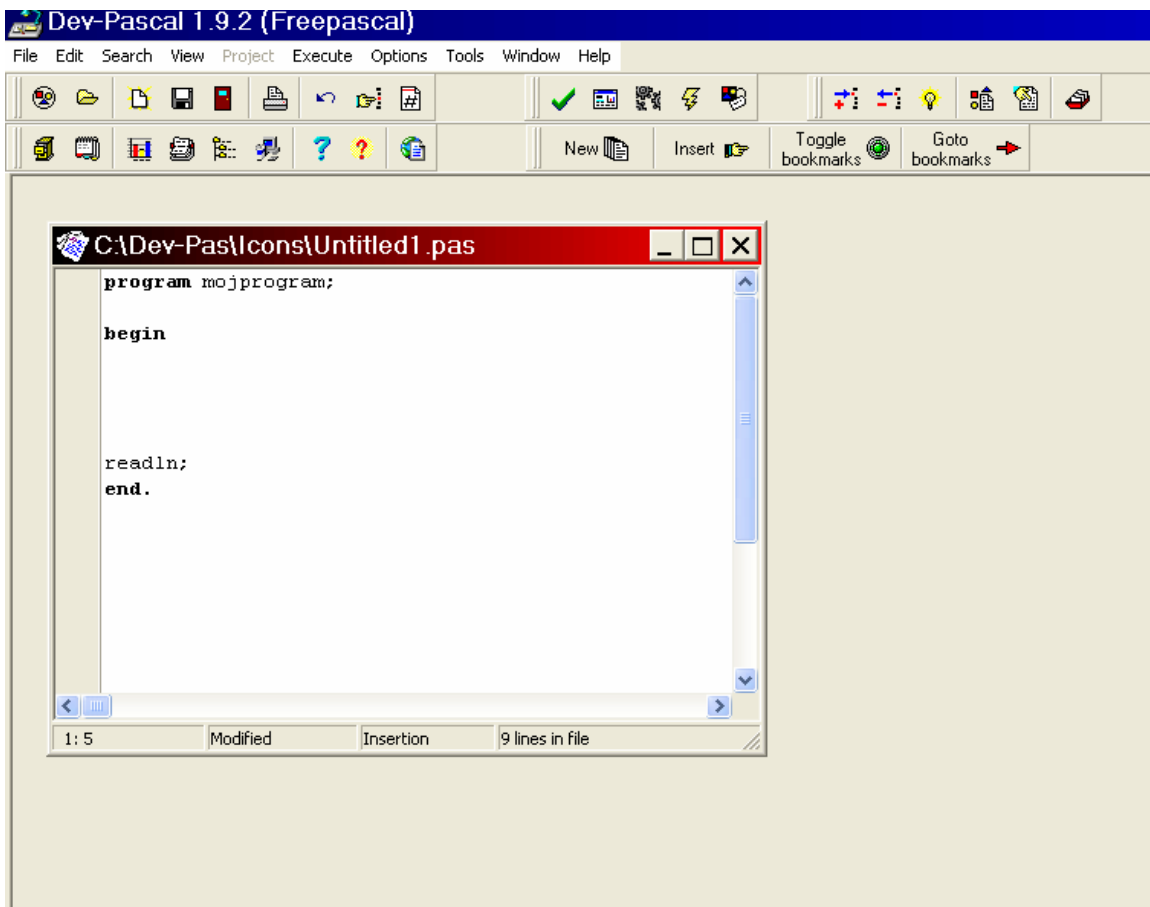
Kostur programa čine rezervisane riječi:

```
program <imeprograma>;  
begin
```

```
readln;  
end.
```

Cijeli program se sada piše između riječi begin i readln.

To bi izgledalo kao na sljedećoj slici:



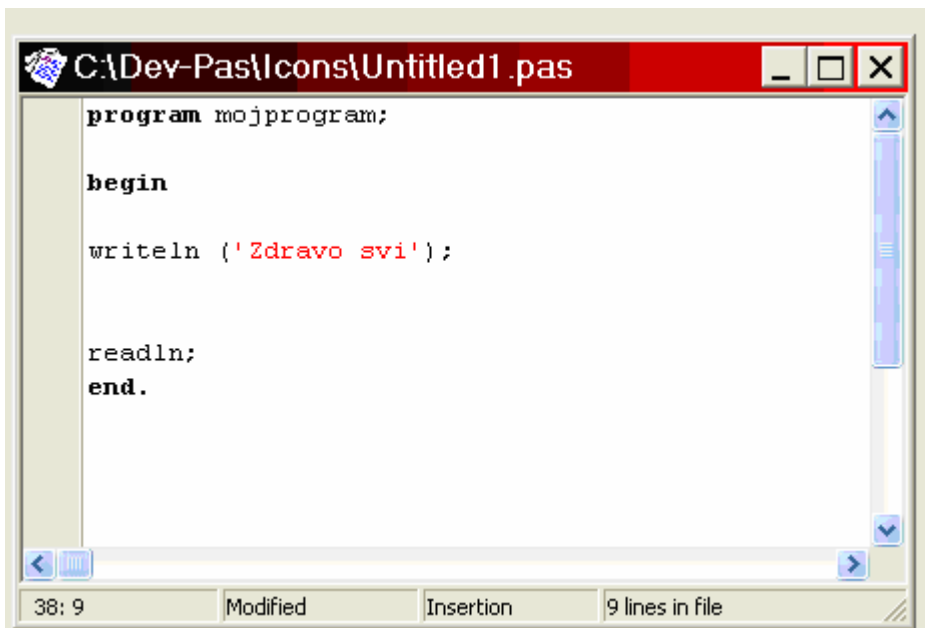
4) Sada smo spremni za pisanje našeg prvog programa

Prva naredba koju ćemo naučiti je naredba **writeln('neki tekst');**

Naredba writeln, ispisuje tekst na ekran.

Tekst mora biti u zagradama, pod jednostrukim navodnicima i na kraju svake naredbe mora biti tačka-zarez.

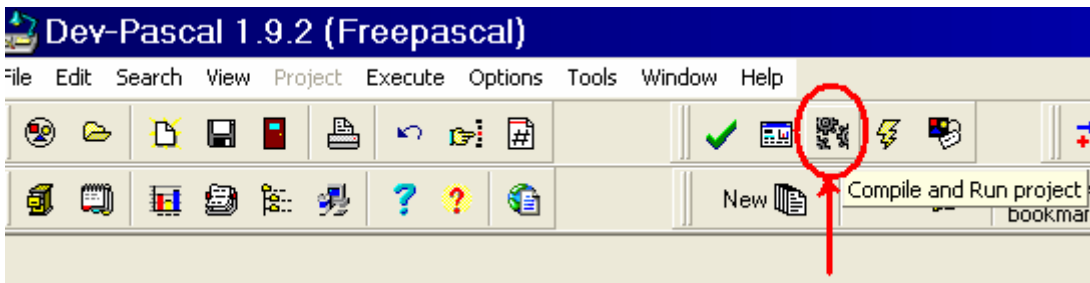
Primjer prikazan na slici:



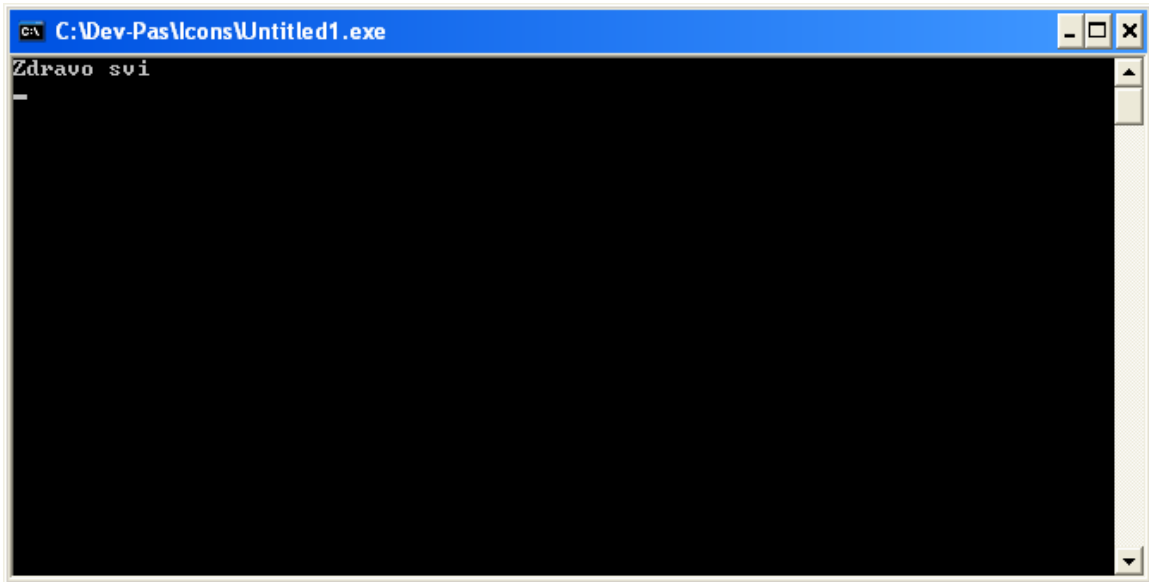
```
program mojprogram;  
  
begin  
  
  writeln ('Zdravo svi');  
  
  readln;  
end.
```

The screenshot shows a window titled 'C:\Dev-Pas\Icons\Untitled1.pas'. The code editor contains the following Pascal code: `program mojprogram;`, `begin`, `writeln ('Zdravo svi');`, `readln;`, and `end.` The status bar at the bottom indicates '38: 9', 'Modified', 'Insertion', and '9 lines in file'.

5) Nakon napisanog programa, potrebno je kompajlirati i pokrenuti program. U tu svrhu koristimo dugme (vidi sliku), koje radi ove dvije navedene operacije.



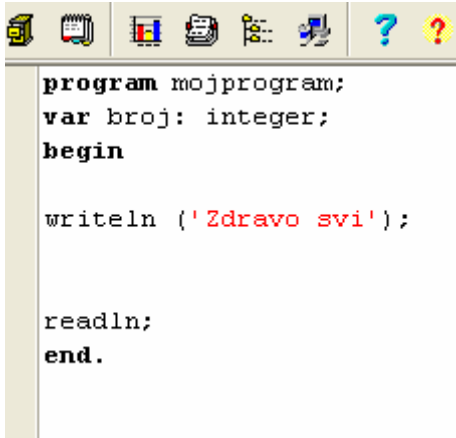
6) Rezultat kompiliranja će biti novootvoreni crni konzolni prozor u kojem se program izvršava.



POGLAVLJE 2: Varijable, writeln i readln naredbe

Varijable

Pojednostavljeno rečeno varijable su **potrebne** kako bi vrijednosti koji dolaze sa **ulaza (tastature ili datoteke)** mogle biti smještene u memoriju i kako bi se mogla vršiti obrada nad istim.



```
program mojprogram;  
var broj: integer;  
begin  
  
  writeln ('Zdravo svi');  
  
  readln;  
end.
```

Kako se vidi na slici gore, varijable se definišu u sekciji **VAR** , koja dolazi odmah iza prve linije programa.

Zatim slijedi **ime varijable**, **dvotačka** te **tip varijable** i **tačka-zarez**.

U našem primjeru ime varijable je **broj**, a sama varijabla je tipa **integer**, tj cijeli broj.

Neki od tipova varijabli su:

- | | |
|----------------|------------------|
| Integer | – cijeli broj |
| Real | – decimalni broj |
| String | – tekst |
| Char | – karakter, znak |

Readln naredba

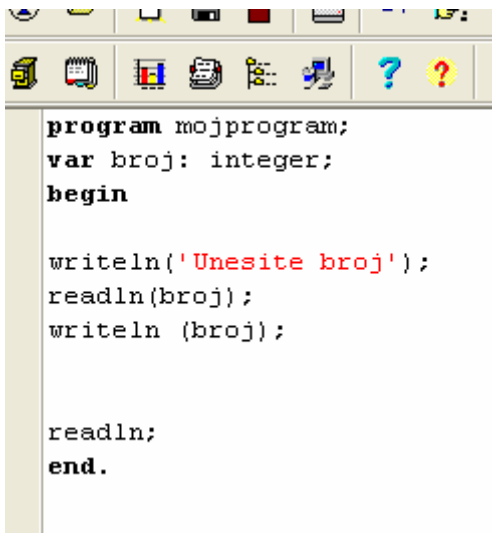
Uz pomoć Readln naredbe moguće je napraviti unos sa tastaure i smjestiti isti u varijablu.

Readln naredba se koristi na sljedeći način **readln(ime_varijable);**

Zadatak 1

Neka korisnik unese broj sa tastaure i neka se taj broj ispiše na ekran.

Rješenje zadatka je dato na slici:



```
program mojprogram;
var broj: integer;
begin

  writeln('Unesite broj');
  readln(broj);
  writeln (broj);

  readln;
end.
```

Pojašnjenje zadatka 1:

U sekciji **VAR** smo definisali varijablu imena **broj** i tipa **integer** (cijelobrojni tip).

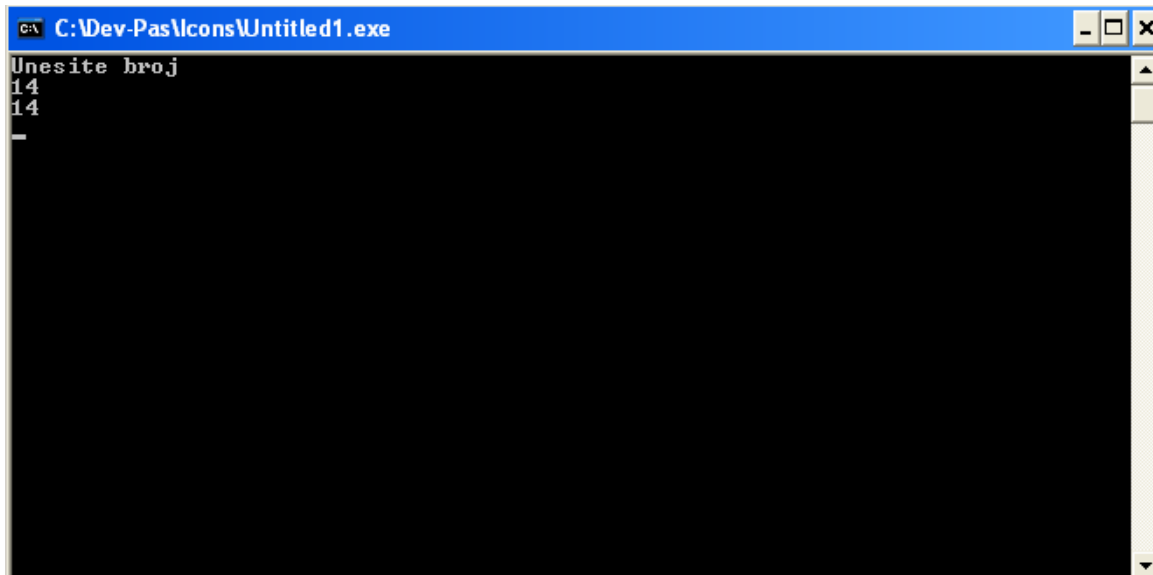
Naredbom **writeln("Unesite broj");** ispisujemo navedeni tekst na ekran, te obavještavamo korisnika da je potrebno unijeti broj.

Zatim smo korištenjem naredbe **readln(broj);** „natjerali“ program da čeka sve dok korisnik ne unese broj, koje će biti smješten u varijablu **broj**.

Sa linijom **writeln (broj);** ispisujemo varijablu **broj** na ekran.

Program je završen.

Izvršenje programa je prikazano na sljedećoj slici:

A screenshot of a Windows console window titled "C:\Dev-Pas\Icons\Untitled1.exe". The window has a black background and white text. The text displayed is: "Unesite broj", "14", "14", and a horizontal line. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

Zadatak 2

Neka korisnik unese 2 broja sa tastaure i neka se zbir ispiše na ekran. !

Rješenje zadatka je dato na slici:

```
program mojprogram;  
var broj1,broj2: integer;  
begin  
  
  writeln('Unesite 2 broja');  
  readln(broj1,broj2);  
  writeln ('Rezultat je:',broj1+broj2);  
  
  readln;  
end.
```

Pojašnjenje zadatka 2:

a) Kako se može vidjeti sa gornje slike, ako trebamo definisati više varijabli istog tipa u **VAR** zaglavlju, potrebno ih je samo nabrojati i odvojiti zarezom:

```
var broj1,broj2:integer;
```

b) Readln parametri se mogu također pobrojati i odvojiti zarezom.

```
readln (broj1,broj2);
```

c) Writeln naredba dopušta unos teksta pod jednostrukim navodnicima , kao i varijabli (**ali bez navodnika !!!**).

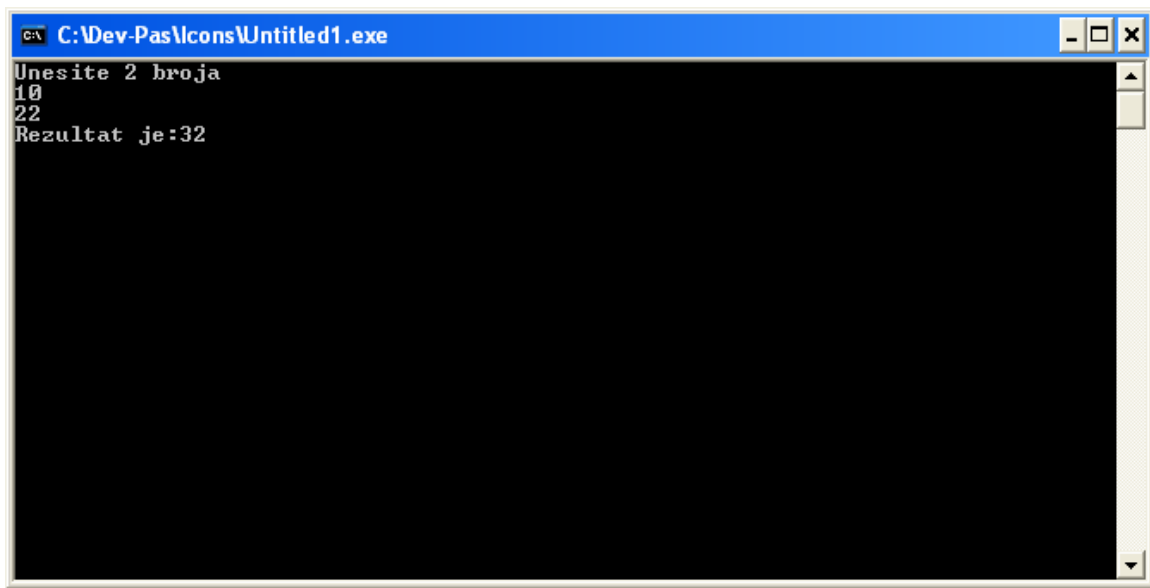
Ako je potrebno ispisati i tekst i varijablu, onda se oni odvajaju zarezom.

```
Writeln ('Rezultat je:', broj1);  
Writeln ('Rezultat je:', broj2);  
Writeln ('Rezultat je:', broj1+broj2);
```

d) Nad varijablama numeričkog tipa (integer ili real) su dozvoljene aritmetičke operacije: sabiranje, množenje... itd. U našem primjeru smo sabrali dvije varijable:

```
Writeln ('Rezultat je:', broj1+broj2);
```

Rezultat kompajliranja:



```
C:\Dev-Pas\Icons\Untitled1.exe  
Unesite 2 broja  
10  
22  
Rezultat je:32
```

POGLAVLJE 3: Iskaz pridruživanja

Najjednostavnije rečeno, iskaz pridruživanje se koristi kako bi varijabli dodjelili ili promjenili vrijednost, a koristi se naredba

:= (dvotačka jednako)

Primjeri:

X:= 2;

U ovom slučaju smo varijabli X dodijelili vrijednost dva

X:= 2+2;

U ovom slučaju smo varijabli X dodijelili vrijednost 4

X:= 5*5;

U ovom slučaju smo varijabli X dodijelili vrijednost 25

X:=Y;

U ovom slučaju smo varijabli X dodijelili vrijednost koju ima i varijabla Y.

X:=X+1;

U ovom slučaju smo varijablu X uvećali za 1;

X:=X*5;

U ovom slučaju smo varijablu X pomnožili sa 5;

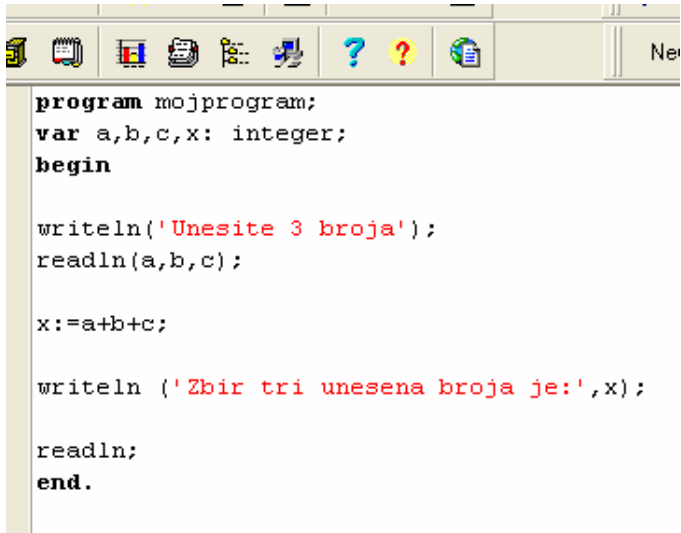
X:=a+b+c+d;

U ovom slučaju smo varijabli X dodijelili zbir varijabli a,b,c i d

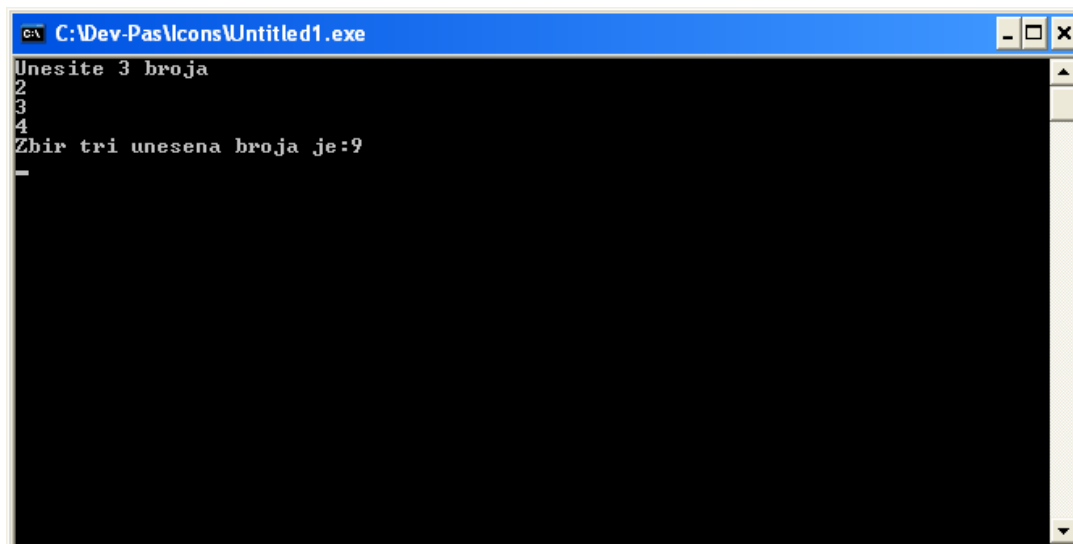
Zadatak 3

Napisati program koji omogućava unos tri cijela broja. Zbir se smješta u varijablu x , a zatim se varijabla x ispisuje na ekran.

Rješenje:



```
program mojprogram;  
var a,b,c,x: integer;  
begin  
  
  writeln('Unesite 3 broja');  
  readln(a,b,c);  
  
  x:=a+b+c;  
  
  writeln ('Zbir tri unesena broja je:',x);  
  
  readln;  
end.
```



```
C:\Dev-Pas\lcons\Untitled1.exe  
Unesite 3 broja  
2  
3  
4  
Zbir tri unesena broja je:9  
-
```

POGLAVLJE 4: Standardne funkcije

Standardne funkcije vrše operaciju nad nekom od varijabli te vraćaju rezultat.

Neke od funkcija su:

<code>sqr(x)</code>	- vraća kvadrat od broja x
<code>sqrt(x)</code>	- vraća korjen od broja x
<code>length(s)</code>	- vraća dužinu riječi s
<code>round(x)</code>	- zaokružuje decimalni broj x na cjelobrojnu vrijednost
<code>abs(x)</code>	- vraća apsolutnu vrijednost broja x

Kod funkcija je bitna napomenuti da **funkcija ne može stajati sama kao naredba:**

Ovo bi bilo pogrešno:

`sqr(x);`

!!! ISPRAVNO JE funkciju:

- dodjeliti varijabli : **`a:= sqr(x);`** (u ovom slučaju varijabli a)
- ispisati na ekran: **`writeln(sqr(x));`**
- ili koristiti za **poređenje**
(npr. u **if** naredbi) **`if (sqr(x)>10) then ...`**

Sve su ovo ispravne metode korištenja funkcija.

Zadatak: 4

Napisati program koji omogućava unos 3 broja. Prvi broj se kvadrira , drugi korjenuje, a od trećeg se ispisuje apsolutna vrijednost.

Rješenje:

Bitno je uočiti da u slučaju za unos broja koji treba korjenovati, potrebno je odgovarajuću varijablu **b** definisati kao decimalni broj, tj kao **real** tip, što se vidi u rješenju.

To je i logično jer korjen cjelog broja ne mora biti nužno biti opet cio broj, dok kvadrat nekog broja je opet cio broj. Stoga je varijabla **a** tipa **integer**.

```
program mojprogram;
var a,c: integer;
b:real;
begin

writeln('Unesite 3 broja');
readln(a,b,c);

writeln ('kvadrat prvog broja je:',sqr(a));
writeln ('korjen drugog broja je:',sqrt(b));
writeln ('apsolutna vrijednost treceg broja je:',abs(c));

readln;
end.
```

```
C:\Dev-Pas\cons\Untitled1.exe
Unesite 3 broja
2
3
-10
kvadrat prvog broja je:4
korjen drugog broja je: 1.7320508075688773E+0000
apsolutna vrijednost treceg broja je:10
```

POGLAVLJE 5: Jednostavni IF ... THEN uslov

Naredbu IF koristimo kada treba da izvršimo poređenje, te na osnovu toga odlučujemo koja naredba će se izvršiti.

Sintaksa IF naredbe je sljedeća:

IF (*uslov*) THEN *naredba*;

Ovo znači ako je ispunjen **uslov**, onda izvrši **naredbu**;

Neki od **uslova** mogu biti sljedećeg tipa:

(a=5)	- ako je varijabla a jednaka broju 5
(a<>5)	- ako je varijabla a različita od broja 5
(a<5)	- ako je varijabla a manja od broja 5
(a<=5)	- ako je varijabla a manja ili jednaka od 5
(a>=5)	- ako je varijabla a veća ili jednaka od 5
(a<b)	- ako je varijabla a manja od varijable b
(a=b)	- ako je varijabla a jednaka varijabli b
(a<>b)	- ako je varijabla a različita od varijable b
(a='f')	- ako je varijabla a jednaka slovu f
(a='mis')	- ako je varijabla a jednaku riječi mis

Primjeri:

IF (a=5) then writeln (a);

Ako je varijabla a jednaku broju 5, onda ispiši na ekran broj a

IF (a=b) then writeln('Varijable a i b su jednake');

Ako su varijable a i b jednake ispiši poruku da su jednake

IF (a<5) then x:=x+a;

Ako je varijabla a manja od 5 onda varijablu x uvećaj za vrijednost varijable a

IF (a>=1) then writeln (sqrt(a));

Ako je varijabla a veća ili jednaka 1 onda ispiši na ekran korjen broja a

Zadatak 5:

Neka se unesu tri broja sa tastature. Ako je zbir unesenih brojeva manji od 10 ispisati kvadrat tog zbira.

Rješenje:

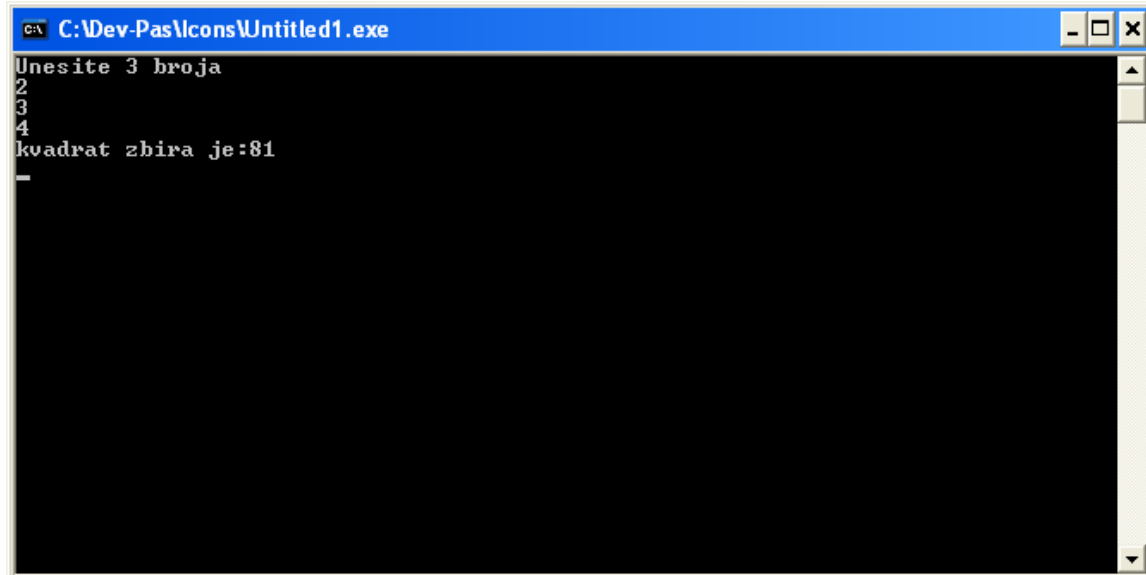
```
program mojprogram;
var a,b,c,zbir: integer;
begin

writeln('Unesite 3 broja');
readln(a,b,c);

zbir:=a+b+c;

if (zbir<10) then writeln ('kvadrat zbira je:',sqr(zbir));

readln;
end.
```



```
C:\Dev-Pas\cons\Untitled1.exe
Unesite 3 broja
2
3
4
kvadrat zbira je:81
-
```


POGLAVLJE 5: Blok na primjeru IF ... THEN

Često se javlja potreba pri korištenju IF THEN naredbe da se izvrši istovremeno više naredbi.

U tom slučaju koristimo **blok** komandu, koja se sastoji od **begin** i **end**.

Primjer:

U slučaju da je varijabla **a** veća od broja 5 potrebno je ispisati dva reda. U prvom redu treba da se ispiše varijabla **a**, a u drugom redu poruka 'broj je manji od 5'

Da bi to uspjeli potrebno je sve naredbe koje je potrebno izvršiti staviti u blok, koji počinje sa **begin** a završava sa **end**

Rješenje bi bilo:

```
IF (a>5) then  
begin  
writeln (a);  
writeln ('broj je manji od 5');  
end;
```

Ovo znači ako je **a** veće od 5 , da će se ispisati obadvije writeln naredbe

Da slučajno nema bloka program bi izgledao ovako:

```
IF (a>5) then  
writeln (a);  
writeln ('broj je manji od 5');
```

U ovom slučaju IF uslov bi vrijedio samo za prvi writeln, te bi drugi writeln bio ispisivan u svakom slučaju bez obzira kolika je vrijednost varijable **a**.

Tj. IF se ne bi uopće odnosio na drugi writeln.

Zato je potrebno, ako ima više naredbi koje se obavljaju na zadani uslov, sve naredbe staviti u blok, kako bi iste bile i izvršene za dati IF uslov.

Zadatak 6:

Neka se unesu 3 broja sa tastature. Ako je proizvod brojeva jednak broju 16, treba ispisati tri reda sa porukom 'BRAVO'

Rješenje:

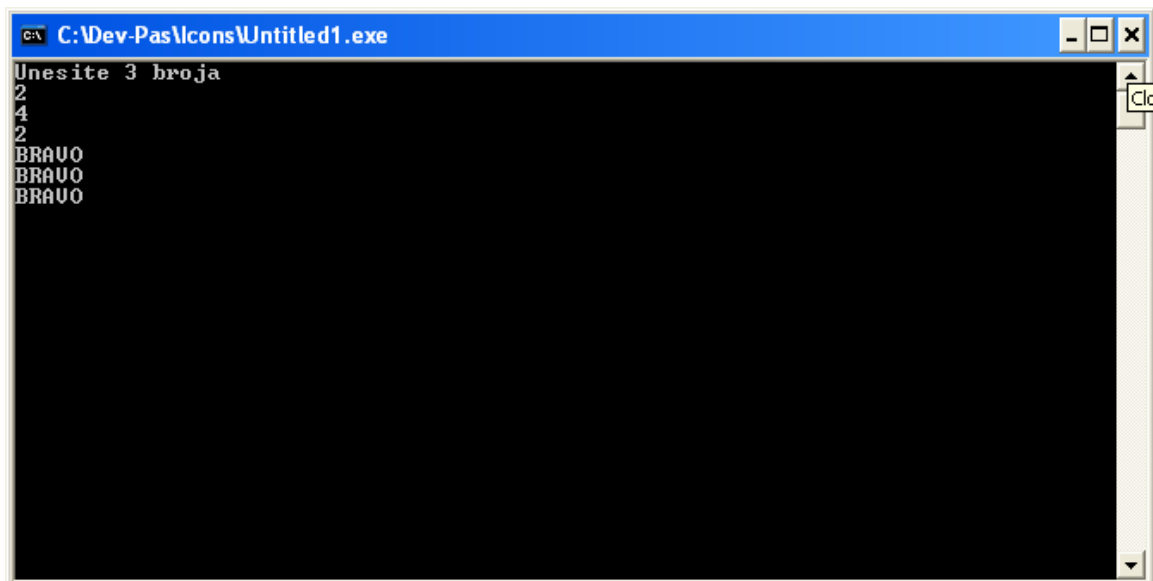
```
program mojprogram;
var a,b,c,produkt: integer;
begin

writeln('Unesite 3 broja');
readln(a,b,c);

produkt:=a*b*c;

if (produkt=16) then
begin // Pocetak bloka (komentar)
writeln ('BRAVO');
writeln ('BRAVO');
writeln ('BRAVO');
end; // Kraj bloka (komentar)

readln;
end.
```



```
C:\Dev-Pas\Icons\Untitled1.exe
Unesite 3 broja
2
4
2
BRAVO
BRAVO
BRAVO
```

POGLAVLJE 7: AND i OR na primjeru IF ... THEN

Nekad je pri korištenju IF naredbe treba ispitati više uslova istovremeno. Pri tome koristimo rezervisane riječi AND i OR , koji predstavljaju logičke operatore.

Ako su uslovi povezani operatorom AND, onda oba uslova moraju biti ispunjena da bi se izvršila IF naredba.

U slučaju da koristimo OR operater, onda je dovoljno da barem jedan od uslova bude ispunjen.

Primjer:

Ako je varijabla **a** veća od nula i manja od 10 potrebno je koristiti sljedeći uslov

IF (a>0) and (a<10) then naredba

Bitno: Primjetiti da su **oba uslova u zagradama** i da su **spojeni sa AND**

Često se ovakvi uslovi pišu **pogrešno**, kao npr: **IF (0<a<10) Ovo je pogrešno !!!**

Primjer 2:

Ako koristimo operator **OR** onda je dovoljno da barem jedan uslov bude ispunjen.

IF (a=1) OR (a=4) OR (a=6) then naredba

U ovom slučaju naredba će biti izvršena ako je ispunjen barem jedan uslov, tj ako je **a** jednako 1, **ili a** jednako 4 **ili a** jednako 6.

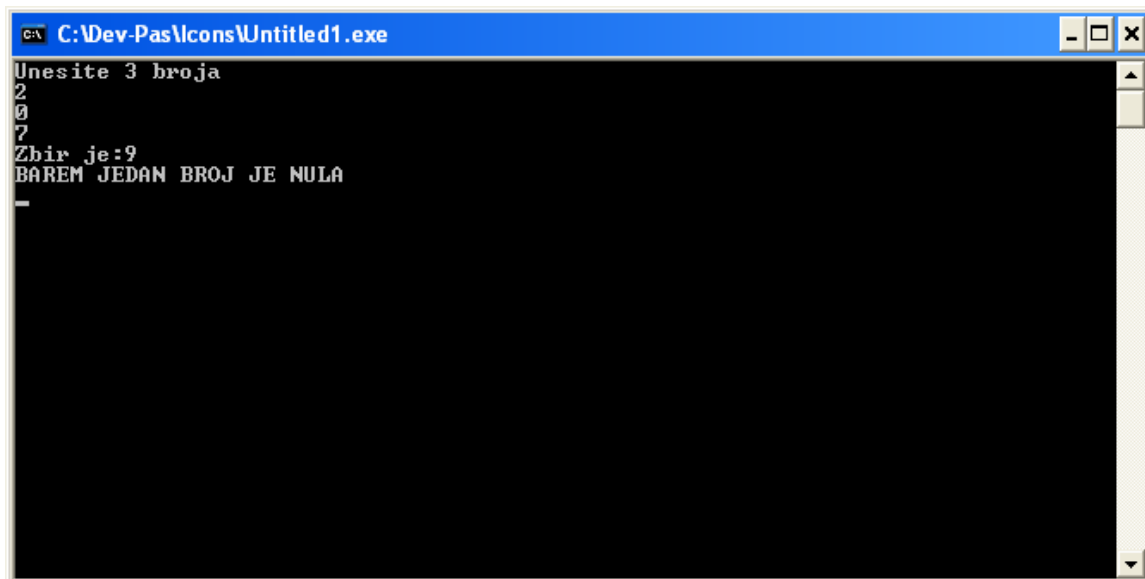
Također i ovom slučaju svaki uslov **mora biti u zasebnim zagradama.**

Zadatak 7:

Napisati program pri kojem se unose tri broja, ako je bilo koji od unesenih brojeva jednak 0 neka program ispiše 2 linije. Prva linija pokazuje zbir svih brojeva . Druga linija ispisuje riječ „BAREM JEDAN BROJ JE NULA“

Rješenje:

```
program mojprogram;  
var a,b,c,zbir: integer;  
begin  
  
  writeln('Unesite 3 broja');  
  readln(a,b,c);  
  
  if (a=0) OR (b=0) OR (c=0) then  
  begin  
    zbir:=a+b+c;  
    writeln ('Zbir je:',zbir);  
    writeln ('BAREM JEDAN BROJ JE NULA');  
  end;  
  |  
  readln;  
end.
```



```
C:\Dev-Pas\lcons\Untitled1.exe  
Unesite 3 broja  
2  
0  
7  
Zbir je:9  
BAREM JEDAN BROJ JE NULA  
-
```

POGLAVLJE 8: IF ... THEN ... ELSE

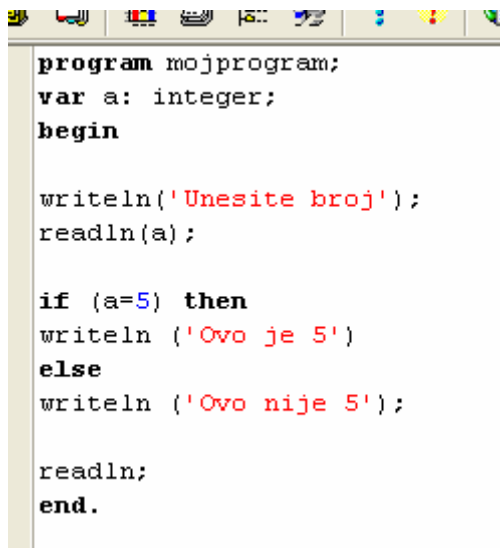
Naredba IF može se koristiti samo ako IF...THEN , ali postoji i opcija **IF ... THEN.. ELSE** koja može pojednostaviti pravljenje programa.

U suština **IF THEN ELSE** funkcioniše na isti način kao i **IF THEN**, međutim postoji mogućnost da se naredba nastavi sa riječi **ELSE** poslije koje slijedi naredba ili blok naredbi u slučaju da početni uslov nije ispunjen.

Primjer:

Ako je varijabla **a** jednaku broju 5, ispiši poruku „Ovo je 5“, u protivnom ispiši poruku „Ovo nije 5“

Rješenje:



```
program mojprogram;
var a: integer;
begin

writeln('Unesite broj');
readln(a);

if (a=5) then
writeln ('Ovo je 5')
else
writeln ('Ovo nije 5');

readln;
end.
```

NAPOMENA: Primjetiti da **poslije** naredbe **writeln('Ovo je 5')**

NEMA tačke-zareza, zato jer se na naredbu nadovezuje ELSE.

Stoga tačka-zarez ide tek **na kraj** zadnje naredbe u IF THEN ELSE izrazu.

Nekad je potrebno koristiti blokove, kako u IF dijelu, tako i u ELSE dijelu naredbe. Vidi naredni zadatak 8 !!!

Zadatak 8: Ako je varijabla a jednaku broju 5, ispiši 2 puta poruku „Ovo je 5“, u protivnom ispiši 2 puta poruku „Ovo nije 5“.

BITNO : U rješenju primjetiti da, **ako se koriste blokovi, svaka naredba u bloku se završava sa tačka zarez**, dok se tačka zaraz za naredbu IF pomjera na zadnji END bloka.

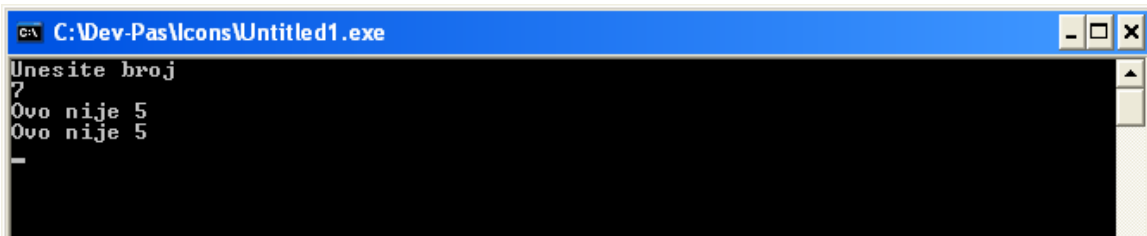
Rješenje:

```
program mojprogram;
var a: integer;
begin

writeln('Unesite broj');
readln(a);

if (a=5) then
begin
writeln ('Ovo je 5');
writeln ('Ovo je 5');
end
else
begin
writeln ('Ovo nije 5');
writeln ('Ovo nije 5');
end;

readln;
end.
```



```
C:\Dev-Pas\lcons\Untitled1.exe
Unesite broj
7
Ovo nije 5
Ovo nije 5
-
```

POGLAVLJE 9: IF grananje

Pod IF grananjem smatramo situaciju, kada kao ELSE naredbu stavimo novi IF uslov, koji opet može da se grana (po potrebi).

Kada se koristi grananje ?

Koristi se ako imamo više uslova koje treba ispitati, a svi uslovi su u podređenom odnosu respektivno, tj neki uslovi su većeg prioriteta od ostalih.

Zadatak 9:

Neka se provjera da li je uneseni broj a jednak broju -5.

Ako jeste jednak -5, onda se ispisuje odgovarajuća poruka i **drugi slučajevi se ne razmatraju**.

Ako nije -5 onda se dalje provjerava da li je uneseni broj negativan.

Ako jeste negativan, ispisuje se poruka, a **drugi slučajevi se ne razmatraju**, a ako a nije ni negativan, onda se ispisuje poruka da je broj pozitivan.

U ovom slučaju najveći prioritet ima uslov da li je broj -5, a ako to nije slučaj, tek onda se ispitivaju ostali negativni brojevi.

Rješenje:

```
program mojprogram;
var a: integer;
begin

  writeln('Unesite broj');
  readln(a);
  |
  if (a=-5) then
    writeln ('Ovo je -5') else
  if (a<0) then
    writeln ('Ovo je negativan broj') else
    writeln ('Ovo je pozitivan broj');

  readln;
end.
```

POGLAVLJE 10: CASE naredba

Case naredba omogućava da se zavisno od datog uslova izvršavaju različite komande.

Sve što se može uraditi sa komandom CASE, može se uraditi i sa komandom IF i obratno.

Sintaksa naredbe CASE je:

```
case varijabla of
    slučaj1: naredba 1;
    slučaj2: naredba 2;
    ...
    slučaj9: naredba 9;
end
else
    slučaj10: naredba 10;
end;
```

Primjer:

Slijedi primjer programa koji provjerava da li je uneseni broj 1, 2 ili neki treći.

```
case a of
1: writeln ('a je 1');
2: writeln ('a je 2');

else

writeln('a nije 1');
writeln('a nije vala ni 2');

end;
```

Primjetiti da nema blokova, jer CASE syntaxa to ne zahtjeva, **ALI** ako bi stavili blok u **ELSE** dijelu, **to ne bi bilo pogrešno** i program bi radio ispravno !!!

Zadatak 10:


Potrebno je unijeti numeričku ocjenu učenika, te ispisati tekst ocjene. Ako ocjena nije u rasponu 1-5 također ispisati odgovarajuću poruku.

Rješenje:

```
program mojprogram;
var a: integer;
begin
    writeln('Unesite ocjenu');
    readln(a);

    case a of
        1: writeln ('nedovoljan 1');
        2: writeln ('dovoljan 2');
        3: writeln ('dobar 3');
        4: writeln ('vrlo dobar 4');
        5: writeln ('odlican 5');
    else
        writeln('Ocjena mora biti od 1 do 5');
    end;

    readln;
end.
```



```
C:\Dev-Pas\Icons\Untitled1.exe
Unesite ocjenu
4
vrlo dobar 4
```

Zadatak 10b:

Korisnik treba unijeti slovo sa tastature. Koristeći CASE naredbu program treba prepoznati da li je uneseno slovo samoglasnik ili suglasnik te ispisati odgovarajuću poruku.

Rješenje:

```
program mojprogram;
var s: char;
begin

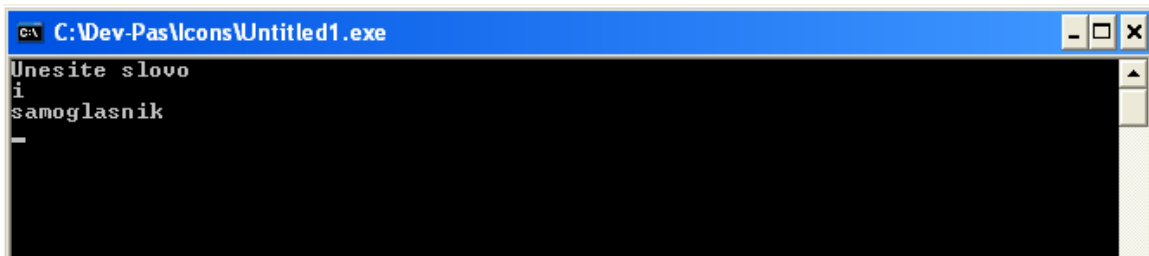
writeln('Unesite slovo');
readln(s);

        case s of
                'a': writeln ('samoglasnik');
                'e': writeln ('samoglasnik');
                'i': writeln ('samoglasnik');
                'o': writeln ('samoglasnik');
                'u': writeln ('samoglasnik');

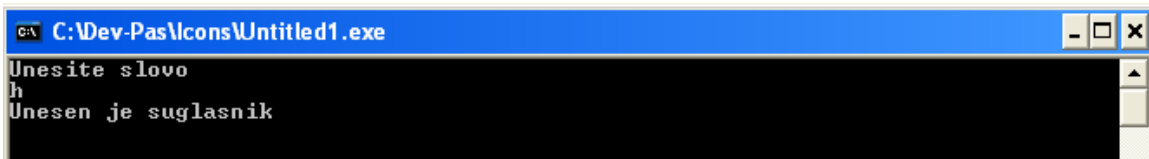
        else
                writeln('Unesen je suglasnik');

        end;

readln;
end.
```



The screenshot shows a command prompt window titled "C:\Dev-Pas\Icons\Untitled1.exe". The prompt displays the text "Unesite slovo" followed by the user input "i". The program's output is "samoglasnik".



The screenshot shows a command prompt window titled "C:\Dev-Pas\Icons\Untitled1.exe". The prompt displays the text "Unesite slovo" followed by the user input "h". The program's output is "Unesen je suglasnik".

KRAJ PRVOG DIJELA