

## 16 Felsorolt típus, record, set típusok

- felsorolt típus
- **record** típus
- **set** (halmaz) típus

### 16.1 Felsorolt típus

A programozásban az eddig felsorolt típusokon kívül (integer, string, byte, boolean, char, array...) használhatunk saját adattípusokat is. Ezeket először a **type** utasítással meg kell adnunk. Az egyik ilyen típus a felsorolt típus, amely csak a felsorolt értékek valamelyikét veheti fel. Pl.

```
program Pelda49;
type nagybetu = 'A'..'Z';
   szamjegy = 0..9;
   nap = (hetfo, kedd, szerda, csutortok, pentek,
         szombat, vasarnap);
var a:nagybetu;
    b:szamjegy;
    n:nap;
begin
  { program }
end.
```

Ezeket a típusokat nem olvashatjuk be közvetlenül a **readln** paranccsal billentyűzetről és nem is írhatjuk ki az értéküket a **write** paranccsal a képernyőre. Beolvasásuk és kiírásuk a következőképpen történhet:

```
program Pelda50;
uses crt;
type nap = (hetfo, kedd, szerda, csutortok, pentek, szombat, vasarnap);
var n:nap;
    i:integer;
begin
  clrscr; {beolvasas}
  writeln('1 - hetfo');
  writeln('2 - kedd');
  writeln('3 - szerda');
  writeln('4 - csutortok');
  writeln('5 - pentek');
  writeln('6 - szombat');
  writeln('7 - vasarnap');
  write('Addj meg egy szamot (1-7): ');
  readln(i);
  case i of
    1: n:=hetfo;
    2: n:=kedd;
    3: n:=szerda;
    4: n:=csutortok;
    5: n:=pentek;
    6: n:=szombat;
    7: n:=vasarnap;
  end; {kiiras}
  write('Az altalad megadott nap a ');
  case n of
    hetfo: writeln('hetfo. ');
    kedd:  writeln('kedd. ');
    szerda: writeln('szerda. ');
    csutortok: writeln('csutortok. ');
    pentek: writeln('pentek. ');
    szombat: writeln('szombat. ');
    vasarnap: writeln('vasarnap. ');
  end;
  readln;
end.
```

## 16.2 Record típus

A **record** típus mezőkből áll, melyek egyedi változóként kezelhetők. Használatát példán szemléltetjük:

```

program Pelda51;
uses crt;
type személy = record
    nev: string;
    szul_ev: integer;
end;
var a: array [1..5] of személy;
    i, jelen: integer;
function beolvas: személy;
var x: személy;
begin
    write('Nev: ');
    readln(x.nev);
    write('Születési év: ');
    readln(x.szul_ev);
    beolvas:=x;
end;
begin
clrscr;
for i:=1 to 5 do
begin
    writeln('Kerem az ', i, '. személy adatait...');
    a[i]:=beolvas;
end;
write('Milyen évet írunk most? (pl.2004) :');
readln(jelen);
for i:=1 to 5 do
    writeln(a[i].nev, ' ', jelen-a[i].szul_ev, ' éves. ');
readln;
end.

```

## 16.3 Set (halmaz) típus

A halmaz típus (**set of ...**) használatát is egy példa segítségével mutatjuk be.

A feladatunk az volt, hogy egy beolvasott szövegben felhasznált betűket írjuk ki ABC sorrendben, minden betűt csak egyszer és minden betűnek a nagybetűs formája legyen kiírva. A feladathoz definiáltunk egy halmazt, amelyet először beállítottunk üresre, majd a betűket sorban beleraktuk ebbe a halmazba. Végül egy **for** ciklus segítségével végignéztük, hogy az ABC melyik betűje van ebben a halmazban (**in**). Ha az adott betű benne volt a halmazban, kiírtuk a képernyőre.

```

program Pelda52;
uses crt;
const betu = ['A'..'Z', 'a'..'z'];
type nagybetu = 'A'..'Z';
var h: set of nagybetu;
    s: string;
    i: integer;
    ch: char;
begin
clrscr;
write('Írj be egy mondatot: ');
readln(s);
h:=[];
for i:=1 to length(s) do
    if s[i] in betu then h:=h+[uppercase(s[i])];
write('Felhasznált betűk:');
for ch:='A' to 'Z' do
    if ch in h then write(ch:2);
writeln;
readln;
end.

```