

## 6 Az IF feltételvizsgálat

- az **IF ... THEN** parancs
- a feltétel **ELSE** ága

### 6.1 Az IF ... THEN parancs

Gyakran előfordul, hogy programunkban valamilyen feltételtől függően szeretnénk parancsot végrehajtani. Például akkor szeretnénk végrehajtani valamilyen utasítást, ha az *i* változó értéke nagyobb mint **5**. Erre általánosan a következő utasítás szolgál:

```
if feltétel then parancs;
```

A mi példánkban (ha *i* nagyobb mint 5) ez így nézne ki:

```
if i>5 then parancs;
```

A feltételben bármilyen két kifejezést összehasonlíthatunk (pl. az *i*+1-et a *j*+5-tel). A kifejezések összehasonlítására használhatjuk a <, >, =, <=, >=, <> (nem egyenlő) jeleket, melyek igaz/hamis (true/false) logikai értékeket adnak vissza. Ha ez a logikai érték igaz (true), akkor a parancs végrehajtódik.

A logikai értékeket összekapcsolhatjuk az **AND** (és), **OR** (vagy), **NOT** (nem), **XOR** (kizáró vagy) műveletekkel. Így egyszerre több feltételt is vizsgálhatunk. Ilyenkor a feltételeket, amelyeket összekapcsoljuk, zárójelbe tesszük. Pl. ha azt szeretnénk, hogy a parancsunk csak akkor hajtsódjon végre, ha az *i* változó értéke nagyobb mint **5** és egyidejűleg a *k* változó értéke egyenlő **7**-tel:

```
if (i>5) and (k=7) then parancs;
```

Természetesen itt is, hasonlóan mint a **for** ciklusnál egyszerre több parancsot is végrehajthatunk a feltételen belül. Ekkor a parancsokat a **begin ... end**; közé rakjuk:

```
if feltétel then begin
    parancs1;
    parancs2;
    ...
    parancsN;
end;
```

**Feladat:** Készítsünk programot, amely bekér három pozitív egész számot, és kiírja őket nagyság szerint növekvő sorrendben!

**Megoldás:** A programban a három számnak 3 változót fogunk használni (**a**, **b**, **c**). Ezeket cserélgetéssel sorba fogjuk rakni úgy, hogy az **a** változóban legyen a legkisebb szám, a **b**-ben a középső és a **c**-ben a legnagyobb szám. A cserékhez egy **x** segédváltozót használunk.

```

program Pelda19;
uses crt;
var a,b,c,x:integer;
begin
  clrscr;
  write('Kerem az elso szamot: ');
  readln(a);
  write('Kerem a masodik szamot: ');
  readln(b);
  write('Kerem a harmadik szamot: ');
  readln(c);
  if a>b then begin
    x:=a;
    a:=b;
    b:=x;
  end;
  if a>c then begin
    x:=a;
    a:=c;
    c:=x;
  end;
  if b>c then begin
    x:=b;
    b:=c;
    c:=x;
  end;
  write('A harom szam novekvo sorrendben: ');
  writeln(a,' < ',b,' < ',c);
  readln;
end.

```

**Feladat:** Készítsünk programot, amely bekér egy mondatot, majd kiírja, hogy hány nagybetű van benne.

**Megoldás:** Egy karakterről úgy tudjuk eldönteni, hogy nagybetű-e, hogy megnézzük nagyobb vagy egyenlő-e mint 'A' és egyben kisebb vagy egyenlő-e mint 'Z'. Ha ez igaz, akkor a karakter A-tól Z-ig van, tehát egy nagybetű. Fogunk használni még egy **nagy** nevű változót, amely a nagybetűk számát fogja számolni.

```

program Pelda20;
uses crt;
var s:string;
    i,nagy:integer;
begin
  clrscr;
  write('Írj be egy mondatot: ');
  readln(s);
  for i:=1 to length(s) do
    if (s[i]>='A') and (s[i]<='Z') then inc(nagy);
  writeln('A mondatban ',nagy,' darab nagy betu van. ');
  readln;
end.

```

A programunkban használtunk egy új függvényt, az **inc()** függvényt (increase). Ez a függvény növeli a változó értékét 1-gyel, tehát az **inc(nagy)** helyett írhattuk volna ezt is: **nagy:=nagy+1**.

Hasonlóan létezik egy függvény, amely csökkenti a változó értékét egygyel, ez a **dec()** függvény (decrease).

## 6.2 A feltétel ELSE ága

A programozás során sokszor előfordul, hogy akkor is végre akarunk hajtani valamilyen utasítást, ha a feltétel nem igaz. Ebben az esetben az **if ... then** parancsunk bővílni fog egy **else** résszel:

```
if feltétel then parancs1
           else parancs2;
```

Ha a feltétel igaz, akkor a *parancs1* hajtódik végre, egyébként a *parancs2* hajtódik végre. Fontos megjegyeznünk, hogy ha az **if** feltételvizsgálatnak van **else** ága, akkor a *parancs1* után nem rakunk pontosvesszőt. Természetesen itt is mindkét parancs helyett tehetünk több egymást követő utasítást is. Ilyenkor minden esetben a **begin ... end** kulcsszavakkal kapcsoljuk össze az egymást követő parancsokat:

```
if feltétel then begin
                parancs1i;
                parancs2i;
                ...
                parancsNi;
            end
           else begin
                parancs1h;
                parancs2h;
                ...
                parancsNh;
            end;
```

**Feladat:** Készítsünk programot, amely bekér két egész számot, majd kiírja szavakkal, hogy az első szám kisebb, nagyobb, vagy egyenlő mint a második szám.

**Megoldás:** A két számot összehasonlítjuk, ha az első szám kisebb mint a második, akkor ezt kiírjuk. Ha ez nem igaz (else), akkor a két számot újra összehasonlítjuk, ha az első szám a nagyobb, akkor kiírjuk, hogy az első szám a nagyobb. Ha ez sem igaz (belső feltétel else ága) akkor pedig kiírjuk, hogy a két szám egyenlő.

```
program Pelda21;
uses crt;
var a,b:integer;
begin
  clrscr;
  write('Kerem az elso szamot: ');
  readln(a);
  write('Kerem a masodik szamot: ');
  readln(b);
  if a<b then writeln('Az 1. szam a kisebb.')
    else if a>b then writeln('Az 1. szam a nagyobb.')
    else writeln('A ket szam egyenlo.');
```

readln;  
end.