

	Algoritmus leíró nyelven	Pascalban
Változó deklaráció	Változó <név>: <típus>	var <név>: <típus>;
Konstans deklaráció	Konstans <név>: <típus> (<érték>)	const <név>=<érték>;
Rekord típus deklaráció	Típus <név>=Rekord(<mezőnév1>: <típus1>, ..., <mezőnév>: <típusn>)	type <név>=Record <mezőnév1>: <típus1>; ...; <mezőnév>: <típusn>; end;
Tömb típus deklaráció	Típus <név>=Tömb[1..<maxindex>]:<típus>]	type <név>=array [1..<maxindex>] of <típus>;
Kiírás	Ki: „<szöveg>”, <változó>	writeln('<szöveg>', <változó>);
Beolvasás	Be: <változó> [<feltétel>]	repeat writeln(<kérdés>); readln(<változó>); until (<feltétel>);
Elágazás	Ha <feltétel> akkor <akkor ág> különben <különben ág> Elágazás vége	if <feltétel> then begin <akkor ág> end else begin <különben ág> end
Számlálós ciklus	Ciklus <cv>=<ké>-től <vé>-ig <ciklusmag> Ciklus vége	for <cv>:=<ké> to <vé> do begin <ciklusmag> end;
Előli tesztelés ciklus	Ciklus amíg <feltétel> <ciklus mag> Ciklus vége	while (<feltétel>) do begin <ciklus mag> end;
Hátul tesztelés ciklus	Ciklus <ciklus mag> amíg <feltétel> Ciklus vége	repeat <ciklus mag> until not(<feltétel>);
Eljárás definiálása	Eljárás <név>(konstans <név>:<típus>, ..., változó <név>:<típus>) változó <név>:<típus> <további konstans, típus, változódeklarációk> <eljárás törzse> Eljárás vége	procedure <név>(const <név>:<típus>; ...; var <név>: <típus>); var <név>:<típus>; begin <eljárás törzse> end;

Függvény definiálása	Függvény <függvénynév> (konstans <név>:<típus>, ..., változó <név>:<típus>):<visszatérési érték típusa> változó <név>:<típus> <további konstans, típus, változódeklarációk> <függvény törzse> <függvénynév>:=<érték> Függvény vége	<pre>function <függvénynév> (const <név>:<típus>; ...; var <név>:<típus>):<visszatérési érték típusa>; var <név>:<típus>; begin <függvény törzse> <függvénynév>:=<érték>; end;</pre>
Program	Program ... <eljárás név>(<paraméter lista>) ... Program vége	<pre>program <név>; uses ...; ... <Konstans-, típus-, változó-, függvény- és eljárás deklarációk> ... begin ... <eljárás név>(<paraméter lista>); ... end.</pre>

Fájlból olvasás

```
<fájl változó>:text;
<fájl név>:string;
...
assign(<fájl változó>;<fájl név>)
reset(<fájl változó>);
...
read(<fájl változó>, <változó>);
...
close(<fájl változó>);
```

Fájlba írás

```
<fájl változó>:text;
<fájl név>:string;
...
assign(<fájl változó>;<fájl név>)
rewrite(<fájl változó>);
...
write(<fájl változó>, <változó>);
...
close(<fájl változó>);
```

Fájlok kezelésével kapcsolatos lekérdezések

```
<vége logikai változó>:=eof(<fájl változó>);
<hiba kód>:=IOResult;
```

Algoritmikus szerkezet	Pascal-szerkezet
Egyszerű típusok	
Egész Valós Szöveg Logikai Karakter	
Típusdefiníció	
Konstans MaxN:Egész (100) Típus TMag=Tömb[1..MaxN:Telem] TDátum=Rekord(év, hó, nap:Egész)	
Változódeklaráció	
Változó N, i:Egész	
Főprogram	
Program Beolvasás(N, X) Feldolgozás(N, X, VANE) Kiírás(VANE) Program Vége	
Adatbeolvasás	
Be: N [0≤N≤MaxN]	
Elágazások	
Ha i>5 és j<3 akkor <utasítás1> <utasítás2> ... Elágazás Vége	
Ha i=10 akkor <utasítás1> különben <utasítás2> Elágazás Vége	

<p>Elágazás</p> <p> i<0 esetén <utasítás1></p> <p> i=0 esetén <utasítás2></p> <p> i>0 esetén <utasítás3></p> <p>Elágazás Vége</p>	
Ciklusok	
<p>Ciklus amíg i<=N és X(i)>100</p> <p> <utasítások></p> <p>Ciklus Vége</p>	
<p>Ciklus</p> <p> <utasítások></p> <p>amíg i<=N</p> <p>Ciklus Vége</p>	
<p>Ciklus i=1-től N-ig</p> <p> <utasítások></p> <p>Ciklus Vége</p>	
Eljárás	
<p>Eljárás Valami (Konstans N: Egész, X: TX, Változó VANE: Logikai)</p> <p>Változó i: Egész</p> <p> <utasítások></p> <p>Eljárás Vége</p>	
Függvény	
<p>Függvény Valami (Konstans N: Egész, X: TX):</p> <p>Logikai</p> <p>Változó i: Egész</p> <p> <utasítások></p> <p> Valami := Igaz</p> <p>Függvény Vége</p>	

Algoritmikus szerkezet	Pascal-szerkezet
Egyszerű típusok	
Egész Valós Szöveg Logikai Karakter	byte, word, shortint, integer real string, string[20] boolean char
Típusdefiníció	
Konstans MaxN:Egész (100) Típus TMag=Tömb[1..MaxN:Telem] TDatum=Rekord (év, hó, nap:Egész)	Const MaxN=100; Type TMag=Array[1..MaxN] of TElem; TDatum=Record ev, ho, nap:Word; End;
Változódeklaráció	
Változó N, i:Egész	Var N, i:Integer;
Főprogram	
Program Beolvasás(N, X) Feldolgozás(N, X, VANE) Kiírás(VANE) Program Vége	program valami; uses crt; ... begin Beolvasas(N, X); Feldolgozas(N, X, VANE); Kiiras(VANE); end.
Adatbeolvasás	
Be: N [0≤N≤MaxN]	Repeat Write('Mi a szösz:'); Readln(N); Until (N>=0) and (N<=MaxN);
Elágazások	
Ha i>5 és j<3 akkor <utasítás1> <utasítás2> ... Elágazás Vége	if (i>5) and (j<3) then begin <utasítás1>; <utasítás2>; ... end;
Ha i=10 akkor <utasítás1> különben <utasítás2> Elágazás Vége	if i=10 then begin <utasítás1>; end else begin <utasítás2>; end;

<p>Elágazás</p> <p> i<0 esetén <utasítás1></p> <p> i=0 esetén <utasítás2></p> <p> i>0 esetén <utasítás3></p> <p>Elágazás Vége</p>	<pre>if i<0 then begin <utasítás1>; end else if i=0 then begin <utasítás2>; end else begin <utasítás3>; end;</pre>
Ciklusok	
<p>Ciklus amíg i<=N és X(i)>100</p> <p> <utasítások></p> <p>Ciklus Vége</p>	<pre>while (i<=N) and (X(i)>100) do begin <utasítások>; end;</pre>
<p>Ciklus</p> <p> <utasítások></p> <p>amíg i<=N</p> <p>Ciklus Vége</p>	<pre>repeat <utasítások>; until not (i<=N);</pre>
<p>Ciklus i=1-től N-ig</p> <p> <utasítások></p> <p>Ciklus Vége</p>	<pre>for i:=1 to N do begin <utasítások>; end;</pre>
Eljárás	
<p>Eljárás Valami (Konstans N: Egész, X: TX, Változó VANE: Logikai)</p> <p>Változó i: Egész</p> <p> <utasítások></p> <p>Eljárás Vége</p>	<pre>procedure Valami(const N: integer; const X: TX; var VANE: boolean); var i: integer; begin <utasítások>; end;</pre>
Függvény	
<p>Függvény Valami (Konstans N: Egész, X: TX):</p> <p>Logikai</p> <p>Változó i: Egész</p> <p> <utasítások></p> <p> Valami := Igaz</p> <p>Függvény Vége</p>	<pre>function Valami(const N: integer; const X: TX): boolean; var i: integer; begin <utasítások>; Valami := true; end;</pre>