

1. tétel

- SZINTAKTIKAI SZABÁLYOK

- A program építőelemei: (82-86)
- szimbólumok (; , *)
 - fenntartott szavak (begin, end, program, var)
 - azonosítók (lekér, asics ...)
 - címlet
 - konstansok (3, 4, 5)
 - szövegliterálok ('cassa = ')
 - elválasztójelek (szóköz, megcsoportosítás, sorvégjel)

- Pascal program felépítése: (FÜZET 1) (4-6)

1. programfej (PROGRAM)

2. deklarációs rész

a, unit $\begin{cases} \text{gyári} \\ \text{privát} \end{cases}$ (crt, graph)

USES

b, típusok

TYPE

c, konstansok

CONST

d, címlet

LABEL

e, változók

VAR

f, függvények = RUTIN

FUNCTION, PROCEDURE

3. Főprogram

ASCII : 0-255

számok 48-57

angol ABC nagybetű 65-90
kisbetű 97-122

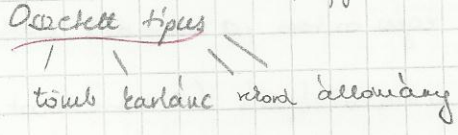
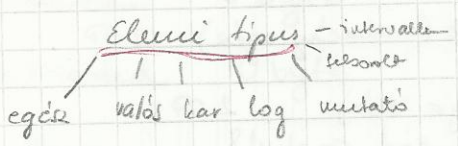
hivatkozás (#...)

ctrl bill : 1-26

2. tétel

(Adatbázis szempontjából) Típusok
 nincs ↑ nevezetű

(elemi v. összetett típusok valamelyike) összefüggés szerint



Egész: előjeles v. nem, 1-4 byte

Tömb: nev: array [1..n] of ...

Valós: 4-10 byte

karakter: STRING

kar: CHAR

rekord: REKORD

logikai: BOOLEAN

állomány

mutató: POINTER

Tömb: olyan homogén adatmennyiség, ahol a memóriát az általában elfoglalt helye azonosítja.

Rekord: olyan heterogén adatmennyiség, amelynek valamennyi komponensére is a rekordra is ötváló névvel hivatkozunk.

Karakter: A változó hosszúságú karakterek sorozatára alkalmazható, és egydim. tömb speciális eseteinek kéntelhető (Hog kell adni a max hosszot!).

(101-118)

EGÉSZ TÍPUS:

- byte (1 byte) var b: byte; (0-255) b:=256 constant out of range
- word (2 byte) 0-65535
- shortint (eger) ^(1 byte) -128-127 (váru absz, kettő kompon + 1.)
- integer ^(2 byte)
- longint ^(4 byte) -2³¹-2³¹-1

Utárelvétel:

NOT (bitenként) 1 OPERANDUSÚ

+, - előjel 1 OP.

* szorzás 2 OP

DIV egész osztás, a maradék elvétel (eredmény egész) 2 OP.

MOD maradékelvétel (a maradékot adja elő) 2 OP.

AND (bitenkénti és) 2 OP

SHL (bitenkénti balra tolás) $ASHL N \rightarrow A$ bitjeit n -nel balra

SHR (bitenkénti jobbra tolás)

+ összeadás

- kivonás

OR (bitenkénti vagy) 2 OP

XOR (kizáró vagy bitenként) 2 OP.

AND

	0	1
0	0	0
1	0	0

OR

	0	1
0	0	1
1	1	1

XOR

	0	1
0	0	1
1	1	0

DEC(i, w) i -t w -vel csökkentés, $w \in \mathbb{Z}, w \leq 0 \Rightarrow i := i - w$

INC(i, w) i -t w -vel növelés, $w \in \mathbb{Z}, w \geq 0 \Rightarrow i := i + w$

PRED(i) i -t 1-gyel csökkentés ($i = 3, \text{pred}(i) \Rightarrow 2$.)

SUCC(i) i -t 1-gyel növelés

ABS(i) i abszolút értékét adja meg.

SQR(i) i^2 -et ad meg ($i \in \mathbb{Z}$)

ROUND(r) r -hez legközelebbi egész értéket adja vissza \rightarrow kerekíti a valódi értéket

TRUNC(r) r egész részét adja meg.

VALÓS TÍPUS:

- single 4 byte
- real 6 byte
- double 8 byte
- extended 10 byte
- comp 8 byte \rightarrow kalkulációs

MANTISSZA $\cdot 10^{\text{exponens}}$

$$0 \leq \text{abs}(\text{mantissa}) < 10$$

exponens előjeles szám (balra vagy jobbra tojhat a szám kerekjelét)

Művelet:

+ , - előjel 1 OP

* szorzás 2 OP

/ valós osztás

+ összerakás

- kivonás

INT(r) fqr: egészrés (a visszaszámított érték valós)

ABS(r) fqr, abszolútérték (" ")

SQR(r) fqr, négyzet (" ")

SQRT(r) fqr, négyzetgyök (" ")

PI fqr, 3,1415926536

RANDOM(határ) fqr, véletlen számot generál [0, határ)

SIN(r) fqr, nívusz ad vissza \rightarrow r: radiánban $\rightarrow \frac{109 \cdot 2\pi}{360}$

LN(r) fqr, termé alapú log

EXP(r) fqr, r exponenciálisát adja vissza e^r

3. tétel

Karakter típus: (CHAR)

4 ASCII kódú karaktert tárolhatunk.

- lehet - karakterkonstans (1 konstans névvel)
- karakter változó

megadható - karakter
- ASCII kód (#12)
- char literal ('A')

FGV-ek: $\text{Ord}(i) \rightarrow$ ASCII kódot ad vissza
 $\text{Chr}(r) \rightarrow$ karaktert ad vissza
 $\text{Upperc}(c) \rightarrow$ nagybetűt ad, ha $"a" \leq c < "z"$

Relációs relációk: kisebb, nagyobb, kisebb az ASCII kóddal
(Értelmezhető: Dec, Inc, Pred, Succ)

Összehasonlítás műveletek:

=, <>, <, >, <=, >=

LOGIKAI TÍPUS (BOOLEAN)

true - false \rightarrow George Boole (XIX. sz.)

Jaró: 1 byte (utolsó bit \Rightarrow true := 1, false := 0);

Műveletek:

AND		
A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

OR		
A	B	A \vee B
0	0	0
0	1	1
1	0	1
1	1	1

XOR		
A	B	A \oplus B
0	0	0
0	1	1
1	0	1
1	1	0

NOT	
A	$\neg A$
0	1
1	0

3. karaktert még \rightarrow bit
SHL, SHR

ODD(i) \rightarrow ha i páratlan \Rightarrow TRUE

ha i páros \Rightarrow FALSE

Felsorolt típus:

megadása:

var

nap: (kedd, szerda, csüt, péntek, szombat, vas);

írás: () ; nincs éé eset [↑] azonosítók

- Ha az azonosító néma $\leq 256 \Rightarrow$ 1 byte-n tárolja

- az első azonosító tárolási kódja 0.

FGV, ELÉ:

DEC, INC, ORD, PRED, SUCC

Művelet:

szorzás \Rightarrow éé eset, amelyiknél éé eset a szám

BOOLEAN is felsorolt típus.

INTERVALLUM TÍPUS (intervalum)

Intervalum típus a programozó adja meg egy már létező sorozatosított

típus intervallumait.

megadása

var hónap : 28..30;

betűk : 'A'..'Z';

FORDÍTÁSI DIREKTÍVÁK \rightarrow intervallumellenőrés

fürtési hiba \Rightarrow "Range check error!"

Tömb:

Önálló adatnév + tömb DEF.

Deklaráció:

tömbnév: array [indexhatár] of elemtípus

Labályok:

- a tömbvet rögzítve néhány elem van
- az elemek azonos típusúak
- index határozható lehet
- index indextípusú
- elemek névén az indextípus névvel a határossal meg
- indextípus minden értékén \exists egy jól meghatározott tömbelem
- az elemek bármelyikre közvetlenül elérhető az indexek valós hivatkozással.

Indexhatár ellenőrzés:

index beleszűrt az indexhatárossal (RANGE CHECK)

Belső adatnév:

- amit a programozó meg kell jegyeznie
- bemenő és kimenő adatnév \rightarrow programozó \Rightarrow ezt a belső adatnév dolgozza fel.

GRAFIKON \rightarrow rétdim-ös tömbtel ...

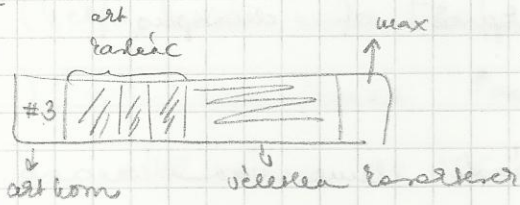
Értéadás

Töbldim-ös tömb

Karakterlánc:

Fogalma:

felépítés



művelet: összekapcsolás operátor, érteladás

érteladás

indexelés:

length: $\text{length}(\text{karakterlánc}) \rightarrow \text{kom.}$

copy: $\text{copy}(\text{karakterlánc}, \text{ésd}, \text{db}) \rightarrow \text{karakterláncot ad vissza}$

pos: $\text{pos}(\text{karakterlánc}, \text{karakterlánc}) \rightarrow \text{karakterlánc első karakterének pos.}$
concat: $\text{concat}(s_1, s_2, \dots, s_n) \rightarrow \text{összeűzés}$

del: $\text{delete}(\text{karakterlánc}, \text{ésd}, \text{db}) \rightarrow \text{törölés}$

insert: $\text{insert}(\text{karakterlánc}, \text{karakterlánc}, \text{pos}) \rightarrow \text{beillesztés}$

str: $\text{str}(\text{ném}, \text{karakterlánc}) \rightarrow \text{némot növeggé}$

val: $\text{val}(\text{karakterlánc}, \text{ném}, \text{ésd}) \rightarrow \text{növeggel némá}$
↓
címke