

1.7

Programozási nyelvek, ea.
SYSTEM UNIT
(folytatás)

Nov. 21.

Stringekkel kapcsolatos fv.-ek:

2. Copy (s: string; kezd, db: integer): string

Hányadik pozíciótól hányadik betűt
kerünk vissza a stringből

pl.: s := copy('almafa', 3, 4); → mafa
('almafa', 3, 14); → mafa
('almafa', 14, 3); → φ

copy('almafa', 1, 3) → alm w

copy(s, length(s)-3+1, 3) → afa w

a string utolsó n betűjét aranyul kiírni:

copy(s, length(s)-n+1, n) w
copy(s, 1, n) g < a f ⊕

delete (var s: string; kezd, db: integer);

A stringből kétszölegesen számú kar.-t töröl
ki, meg kell mondani a pozíciót is!

pl.: s := 'almafa';
delete(s, 3, 2); → alfa

Ha több kar.-t aranyul törölni, mint
amennyiből áll, akkor a string végéig
töröl.

delete(s, 3, length(s)-2) } a string 3. betű-
delete(s, 3, 255) } jétől a végéig
töröl.

length (string, index) -2, -3

Megadja a string aktualis hosszát.

Max. hossz: 255

sizeof (string): milyen hosszú szöveget lehet elhelyezni a stringben.

fgv. pos (mit, miben: string): integer

String bejegyzés kereskedőnek stringet és az az a pozícióját adja vissza. Ha nem talált semmit, 0-t ad vissza.

pl.: S:='12,24,36,3,16';

while length (S) > 0 do begin
i := pos (',', S); \Rightarrow vesszőt keres, de töltés van, így csak az első adja vissza.
if i > 0 then begin ^{szűkítés}

writeln (copy (S, 1, i - 1)) \Rightarrow 12

delete (S, 1, i) \Rightarrow törli a 12-t és a , -t.

end;
else begin
writeln (S);
S := '';
end;
end.

Ha nem aranyul else ágat írni, akkor a string végéhez hozzá kell írni: S := S + ' ';

pos ('a', 'Alma'spité') \Rightarrow 0, mert a nagy A betű \neq a -val

pl.: Insert (mit: string; var miben: string; poz: integer)
A string bejegyzés beszúrásának 1 módja stringet.

almajára

- 3.

insert ('ár', s, 6); \Rightarrow almajára
vagy ugyanezt az eredményt kapjuk, ha:

$S := \text{copy}(s, 1, 5) + \text{'ár'} + \text{copy}(s, 6, 255);$

$\text{length}(s) \equiv \text{ord}(s[\phi])$

$S := \text{'almá'}; X := \text{'acós'}$

$X := s;$

for $i := 1$ to $\text{length}(s)$ do $x[i] = s[i];$

karaktersenként írja fölül az x karaktereit

Eredmény: almácos

Folyamatvezetés:

pr. EXIT;

az azt. alprg. végére ugrik minden átmenet
nélkül; főprg.-ban is használható

if ... then exit; \Rightarrow nem szabad beházi, csak az
else ágba

EXIT: üklelés és vált. \rightarrow belsőben is lehet

pr. HALT (= HALT(\emptyset))

pr. HALT (err: integer)

Azonnal megállítja a prg. futását azon a
ponton, ahol ezt a parancsot kiadta.

Hibakódot lehet paraméterként megadni.

\emptyset hibakód: nincs hiba, rendszeren lefutott

a prg. A hibakódok minden prg.-ban

mások lehetnek

pr. RunError;

pr. RunError (err: integer);

ugyanúgy megállítja a prg.-t, mint a HALT.

Emor (18h) } kirja, mit jelent
a 184-es hibakód
END.

Stringkezelő eljárás (folyt.)

pr. Val (s: string; var x; var c: integer);
Egy stringben megadott névvel a konvertálási
számítási módok: -ra (str → szám)

pl.: var t: integer; r: real;
Val ('123', r, c);
t = 1233 = értelkezt jelembe meg
if c = 0 then r := t
else r := t / 1000;

('12 E 1', r, c)
↓
12 · 10¹

pr. Str (x, var s: string);
Számból stringet csinál.

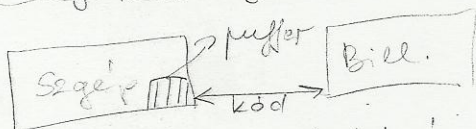
pl.: str (123, s) → s = '123'
str (12.3, s) → s = '1.23 E 01'

↓ számoláshoz konvertálási

str (12.3 : 8 : 2, s) → s = '12.30'
8 kar., 2 tizedesjegy pontossággal
előjellel együtt van, "(.2.30)

ICRT unit

3. fel. képernyő, hangkezelés



Billentyűzet kommunikál a számítógéppel. A bill.-nek van kódja. Puffer fogadja a bill. kódokat, 10 leütésig tárolja a kódokat → a puffert 10 elemű.

Építés: a puffer betölt CRT eljárások a puffert kezelik.

fg: KeyPressed: boolean;

A bill. pufferekben van-e tárolt bill. vagy nem.

Igaz, vagy hamis.

Ha van a pufferekben kód

nincs a pufferekben kód

fg: ReadKey: char

Ha a következő pufferekben várakozó bill. kód kiolvassa a puffer tartalmát és ez is ürti.

var c: char

begin

writeln ('Billentyűleütéssel tovább megy!')

c := readkey;

while KeyPressed do c := readkey;

c := readkey;

itt
effekt
↓
várakozó
puffert

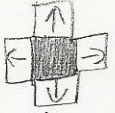
↓
várakozik egy billentyű leütésére

DEL

la

sh alt ctrl

Zeítésűkhöz
semmi sem
fontos



valamelyik
betűt kájjuk
meg, ha leütjük

begin
c: char;
begin

repeat

c := readkey;

writeln('Bill =', c, ' kód, a =', ord(c));

until c = #27; => ESC bill. kódja 27

end:

billja, a kódot a leütött bill.-nel.

pl:

⇒ ' ' φ } két kód kerül
' ' φ } be egyszerre a pufferbe,
az első mindig φ

if c = #φ then begin c := readkey;

EN
\$TER: 13

grafikus ábrázolást a

pascalban

graph^{PH} és $\text{graph}^{\text{PH}} 3$ unit

a GRAPH UNIT:

TURBO PASCAL 5.5

GRAPH.TPU EGAVGA.3GI

és ellenek, ha grafikuson akarunk programozni

a 7-es pascalhoz 7-es kell, az 5.5-ösköz 5.5-ös graph kell.

A graph unit a képernyőt áttevő konverterekből grafikusba, ahol más felbontást produkál

normál esetben: 640×320 pixel

640×480 pixel

Megfigyelhető olyanra, hogy bármelyik gép üzemelője, csak az egészen nem olyan nagy felbontású lesz, mint a monitor.

GRATDEMO.PAS:

Fontos: a használatt változó típusa olyan legyen amilyet a leírás előír.

(nem jeles hibát, csak villan egyet \rightarrow azé a nem működik, mert nem adott típusú a paraméter)

program grafikus;

uses crt, graph;

var a, b, c, x, y, z, x1, x2, y1, y2, meret : integer;

h, i, j, k, l, r, rx, ry, minta, vastag, stilus, font, irany : word

keset : boolean;

begin

clrscr; randomize;

writeln ('GRAFIKAI DEMOPROGRAM');

readln;

DETECTGRAPH (a, b);

INITGRAPH (a, b, 'a');

x := graphresult;

if x < 0 then (ha 0: sikeres)

begin writeln ('A grafikus mod megjelölése sikeres');

writeln ('A hibaról: 'x); readln; halt

end;

{ ----- }

Leptöbbsör olyan kiba van, hogy nem találja a

komponenseket. ↓ grafikus üzembe

ha 1/2-vel, akkor a képméret 1/2-ére csökken

h := GETMAXX div 2; j := GETMAXY 2; → megjelölés felbontás

PUTPIXEL (h, j, WHITE); → ez egy pontot a az x coord. h, és y coord. j -nél helyre. vintő

for i := 1 to 10 do

~~ARC (h, j, 0, 0 + i * 36, 20 + i * 15);~~

ARC (h, j, 0, 0 + i * 36, 20 + i * 15);

↓
körvonal rajzol

10 db körvonal rajzol ki egy nagyobb szögű és körvonal. (le lehet állítani a szög, alapít: körvonal - kör)

meg kell adni a középpontot, a sugarat, hol kezdődjön és végződjön az

ív (°-ban)

