

KÖRRAJZOLÓ RUTIN

Az alábbi kis programmal az EXT utasítással rajzolhatunk köröket.

A kör rajzolása az EXT sorszám, x, y, r utasítással történik, ahol x és y a kör középpontja, r pedig a sugara. Mivel az EXT paraméterei a -32768-tól +32767-ig terjedő tartományba eshetnek, ezért a képernyőn kívül eső középpontú kört is rajzolhatunk! A rutin a kört egy 806 oldalú sokszöggel közelíti, így csak azokat a szakaszokat rajzolja meg, melyeknek mindkét végpontja a képernyőn belül van. Természetesen a rajzolás az aktuális tintaszín, MODE és STYLE paraméterek szerint történik, de a STYLE vonaltípus csak nagyobb átmérőknél érvényesül.

Az első lefuttatás után EXT0,512,480,400 paranccsal próbáljuk ki a rutint. Ha nem kaptunk kört, akkor hiba van a DATA sorokban, javítsuk ki! Ha kört kaptunk, akkor a DATA sorok már feleslegesek, kitorölhetőek a "DELETE 4-" paranccsal. (A gépi kódú program az első sorokba került.) Az így elkészült program magnóra kimenthető, későbbi programokban felhasználható.

Fontos apróságok:

A rutin csak akkor működőképes, ha a szokásos BASIC program kezdőcímről (6639) LOMEM utasítással nem mozdítjuk el. A programot csak a 3. sortól listázzuk, mert az első sorok listázása a többiét is megzavarhatja.

A körrajzoló algoritmus leírása:

Ha az (x;y) pontot az origó körül szöggel elforgatjuk, új koordinátáit (ux;uy) az

$$ux = x * \cos(\varphi) + y * \sin(\varphi)$$

$$uy = y * \cos(\varphi) - x * \sin(\varphi)$$

képletek segítségével számíthatjuk ki. Ha az elforgatást többször megismételjük, és a kapott pontokat összekötjük, sokszöget kapunk. Nem origó középpont esetén a középpont koordinátáit (kx,ky) a kapott pontokhoz hozzá kell adni. A rutin a kört egy 806 oldalú sokszöggel közelíti. Ebben az esetben $\sin(\varphi) \approx 1/128$, $\cos(\varphi) \approx 32767/$

A szerkesztő azért van,

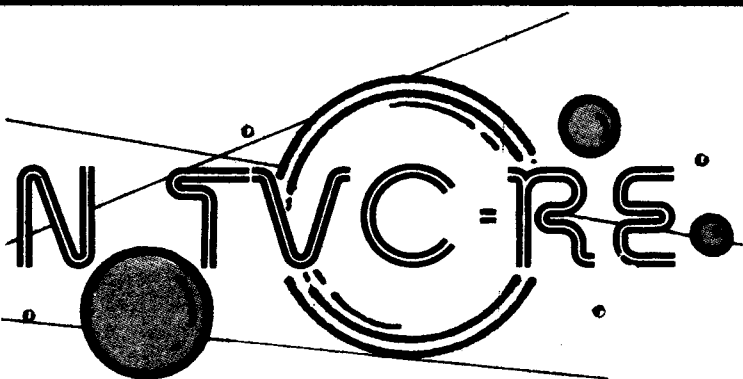
hogy a lap olyan legyen,

amilyenek az olvasói!

```

250          ORG 6639
              DB 250      ; sorhossz
              DW 0        ; sorszám
254          DB 254      ; ! tokenje
              KX:DW 0
              KY:DW 0
              X:FILL 4,0
              Y:FILL 4,0
              UX:FILL 4,0
              UY:FILL 4,0
              VAR:FILL 4,0
              VAR2:FILL 4,0
              MOVE:LD BC,4
237,176      LDIR
201          RET
              MINUS:LD B,4
183          OR A        ; CY:=0
26          M1:LD A (DE)
158          SBC A,(HL)
18          LD (DE),A
35          INC HL
19          INC DE
16,249      DJNZ M1
201          RET
              SIN_LFI:LD B,0
126          LD A,(HL)
23          RLA
48, 1       JR NC,S1
5           DEC B        ; B:=255
43          S1:DEC HL
43          DEC HL
17, 7, 26  LD DE,VAR
213         PUSH DE
229         PUSH HL
197         PUSH BC
205, 15, 26 CALL MOVE
241         POP AF
50, 10, 26 LD (VAR+3),A
225         POP HL
43          DEC HL
126         LD A,(HL)
23          RLA
225         POP HL
203, 22     RL (HL)
35          INC HL
203, 22     RL (HL)
35          INC HL
203, 22     RL (HL)
201         RET
              COS_LFI:LD BC,0
126         LD A,(HL)
23          RLA
48, 1       JR NC,C1
11          DEC BC      ; BC:=65535
43          C1:DEC HL
229         PUSH HL
197         PUSH BC
17, 11, 26 LD DE,VAR2
205, 15, 26 CALL MOVE
193         POP BC
237, 67, 13, 26 LD (VAR2+2),BC
225         POP HL
43          DEC HL
229         PUSH HL
126         LD A,(HL)
23          RLA
33, 11, 26 LD HL,VAR2
203, 22     RL (HL)
35          INC HL
203, 22     RL (HL)
35          INC HL
203, 22     RL (HL)

```



32768. Mindkét értékkel viszonylag könnyen lehet binárisan szorozni. E szorzásokat a SIN_FI és COS_FI szubrutinok végzik az $n/128=2*n/256$ és $32767/32768*n=n-n/32768$ számítások alapján. Az ux, uy, x, y változók és var, var2 részeredmények tárolása 32 biten történik. (16 bit egész rész, 16 bit tört rész.) A középpont kx, ky koordinátáinak tárolása 16 biten történik. Az összeadást és kivonást a PLUS és MINUS rutinok végzik, a változókat a MOVE rutin helyezi a megfelelő helyre.

A kiszámolt pontokhoz a középpont értékének hozzáadását, és a szakasz megrajzolását a PLOT címen kezdődő rész végzi. Sajnos a TVC koordináta rendszerének vízszintes és függőleges léptékezése nem azonos, ezért az x koordináta 7/8-ad részével kell rajzolni. Ezt a számítást a ZSUGOR címen kezdődő rész végzi.

A vonalak meghúzását a TVC operációs rendszerének rutinjai végzik, melyek az RST48 (RST30H) utasítással érhetőek el.

Bata László,

1137 Budapest, Jászai Mari tér 4. I/4.

```
10 !***** / osszesen 245 db. csillag
20 !***** / osszesen 63 db. csillag
30 POKE 33,130: POKE 34,26: REM EXT 0,x,y,r
40 FOR I=6639 TO 6956: READ A: POKE I,A: NEXT
50 DATA
60 DATA
```

(A DATA sorokba az Assembler lista bal oldalán lévő adatokat kell írni!)

```
225 POP HL
43 DEC HL
17, 7, 26 LD DE,VAR
213 PUSH DE
205, 15, 26 CALL MOVE
209 POP DE
33, 11, 26 LD HL,VAR2
205, 21, 26 CALL MINUS
201 RET
247,9 VEG:RST 48,9 ; toll fel
247, 9 VEG:RST 48,9 ; toll fel
251 EI
201 RET ; a program vege
243 KEZDET:DI ; a program eleje
34,243, 25 LD (KX),HL
237, 83,245, 25 LD (KY),DE
237, 67,253, 25 LD (Y+2),BC ; CY:=r
33, 0, 0 LD HL,0
34,251, 25 LD (Y),HL
34,247, 25 LD (X),HL
34,249, 25 LD (X+2),HL
33, 39, 3 LD HL,807; ciklushossz
229 PUSH HL
225 CIKLUS:POP HL
43 DEC HL
17, 0, 0 LD DE,0
183 OR A ; CY:=0
237, 90 ADC HL,DE; HL=0?
40,214 JR Z,VEG ; kesz a kor!
229 PUSH HL
42,245, 25 PLOT:LD HL,(KY)
237, 91,253, 25 LD DE,(Y+2)
25 ADD HL,DE; HL:=y+ky
235 EX DE,HL ; DE:=y koord.
42,249, 25 ZSUGOR:LD HL,(X+2)
68 LD B,H
77 LD C,L
203, 40 SRA B
203, 25 RR C
203, 40 SRA B
203, 25 RR C
203, 40 SRA B
203, 25 RR C ; BC:=HL/8
183 OR A ; CY:=0
237, 66 SBC HL,BC; HL:=7/8 HL
237, 75,243, 25 LD BC,(KX)
9 ADD HL,BC; HL:=x+ky
68 LD B,H
77 LD C,L ; BC:=x koord.
247, 6 RST 48,6 ; pozicionalas
183 OR A ; hiba tortent?
40, 4 JR Z,PL1 ; nem volt hiba
247, 9 RST 48,9 ; toll fel
24, 2 JR PL2
247, 8 PL1:RST 48,8 ; toll le
33,250, 25 PL2:LD HL,X+3
205, 71, 26 CALL COS_FI; var:=x*cos(fi)
33, 7, 26 LD HL,VAR
17,255, 25 LD DE,UX
213 PUSH DE
229 PUSH HL
24, 5 JR UJLSOR+4
255 DB 255 ; sorveg
68 UJLSOR:DB 68 ; sorhossz
0, 0 DW 0 ; sorzam
254 DB 254 ; ! tokenje
205, 15, 26 CALL MOVE; ux:=var
33,254, 25 LD HL,Y+3
205, 32, 26 CALL SIN_FI; var:=y*sin(fi)
225 POP HL
209 POP DE
6, 4 PLUS:LD B,4
183 OR A ; CY:=0
26 P1:LD A,(DE)
142 ADC A,(HL)
18 LD (DE),A
35 INC HL
19 INC DE
16,249 DJNZ P1 ; ux:=ux+var
33,254, 25 LD HL,Y+3
205, 71, 26 CALL COS_FI; var:=y*cos(fi)
33, 7, 26 LD HL,VAR
17, 3, 26 LD DE,UY
213 PUSH DE
229 PUSH HL
205, 15, 26 CALL MOVE; uy:=var
33,250, 25 LD HL,X+3
205, 32, 26 CALL SIN_FI; var:=x*sin(fi)
225 POP HL
209 POP DE
205, 21, 26 CALL MINUS ; uy:=uy-var
1, 8, 0 LD BC,8
33,255, 25 LD HL,UX
17,247, 25 LD DE,X
237,176 LDIR ; x:=ux, y:=uy
195,158, 26 JF CIKLUS
255 DB 255 ; sorveg
```