

PRESEK

List za mlade matematike, fizike, astronome in računalnikarje

ISSN 0351-6652

Letnik 13 (1985/1986)

Številka 1

Strani 38-44

Andrej Vitek:

KAKO RAČUNALNIK NARIŠE DALJICO

Ključne besede: računalništvo, matematika.

Elektronska verzija: <http://www.presek.si/13/747-Vitek.pdf>

© 1985 Društvo matematikov, fizikov in astronomov Slovenije

© 2010 DMFA - založništvo

Vse pravice pridržane. Razmnoževanje ali reproduciranje celote ali posameznih delov brez poprejšnjega dovoljenja založnika ni dovoljeno.

KAKO RAČUNALNIK NARIŠE DALJICO

Očetje in sinovi, pa tudi kakšna hči in celo mamica so si zadnja leta našli novo igračo: hišni računalnik. Pravo malo čudo je takale škatlica, kajne? Sin se z njo podi za Marsovci, očka piše nekakšne službene zadeve, hčerka odkriva skrivnosti in zaklade kraških jam, mamica pa si tu in tam ogleda svoj bioritem. In vsa ta resna in manj resna dogajanja spremlja risba na televizijskem zaslonu: črta kaže, kako raste in pada bioritem, po drugi črti potuje marsovska vesoljska ladja. Le kako zna računalnik vse to narisati?

Vas zanima? Pa si oglejmo!

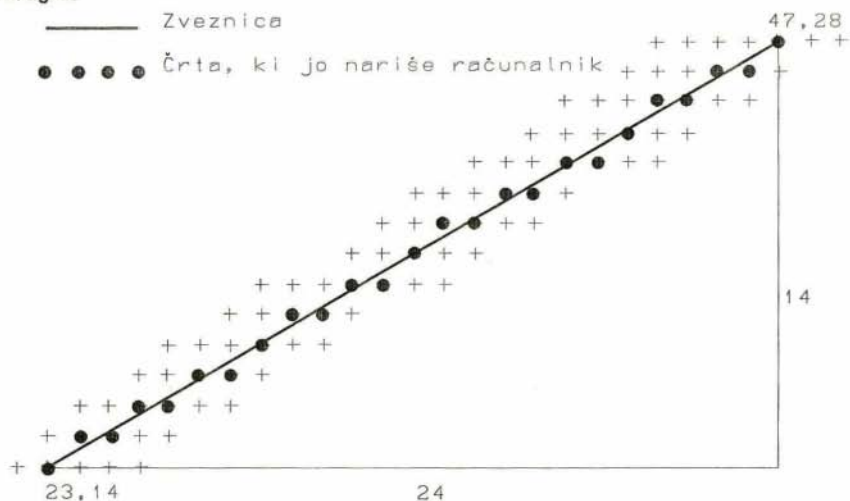
RAČUNALNIKOV PAPIR

Kje bi začeli? Pri samem začetku: oglejmo si, kako računalnik nariše sliko na zaslon. Nekateri med vami vedo, kako sliko riše televizor, drugi se o tem lahko takoj prepričate: prižgite televizor in sliko pogledajte čisto od blizu. Opazili boste, da je slika sestavljena iz množice vodoravnih vrstic. Podobno postopa tudi računalnik, le da so pri njem vrstice razdeljene še na barvne ali črno-bele pike, in to tako, da je zaslon velika rešetka različno pobarvanih pik. Pike stoje druga pod drugo, urejene v stolpce. Računalnikove pike so oštevilčene, zato pač, da lahko pot od pike do pike izračuna. Piko izberemo s parom naravnih števil; prvo število pove stolpec, drugo pa vrstico, v kateri stoji pika. Stolpce in vrstice štejemo od levega spodnjega vogala zaslona. Pika 23,14 je tako triindvajseta pika v štirinajsti vrstici. Paru števil, ki označujeta piko, pravimo s tujko koordinati. Tako torej lahko ukažemo: pobarvaj piko 23,14, ali v basicu: PLOT 23,14. Igranje z barvo in podobnimi učinki pa prepuščam vam, dragi bralci.

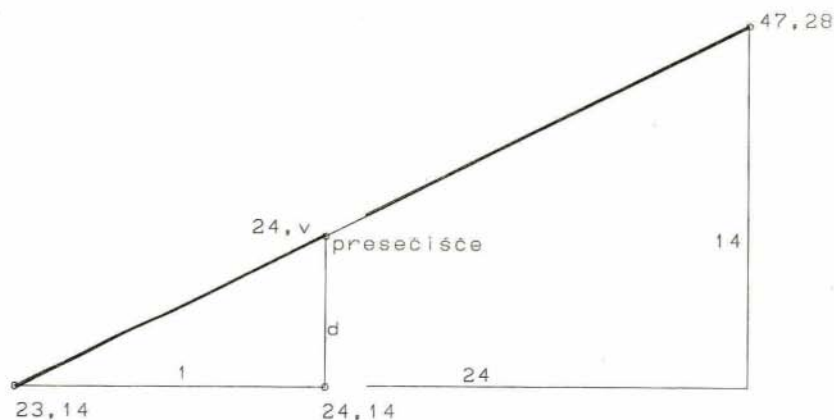
RAČUNALNIKOVO RAVNILO

Sedaj pa k stvari: oglejmo si, kakšno je računalnikovo ravnilo, po kakšnem postopku računalnik nariše ravno črto. Denimo od točke 23,14 do točke 47,28, se pravi 24 pik v desno in 14 navzgor. Nekateri med vami me boste na kratko odpravili: z ukazom DRAW 24,14 pač, boste rekli. Res je, dosti računalnikov

je, ki "znajo" risati ravne črte, pri katerih je postopek za risanje vgrajen kar v računalnik sam. Pa vam utegne priti postopek za risanje ravne črte prav tudi na takem računalniku. Kdaj? Denimo, ko boste v svoji igrici želeli vesoljsko ladjo poslati z enega planeta na drugega – po ravni črti. A več o tem morda kdaj drugič.



Slika 1. Skica naše naloge



Slika 2. Podobni trikotniki

Preden se naloge lotimo, si jo skicirajmo. Na sliko 1 smo narisali kar naš gornji primer, črto od 23,14 do 47,28. Ti piki povezuje črta, potegnjena s pravim ravnilom. Z računalnikom bomo morali pobarvati tudi pike, katerih središča ne leže na tej črti. Črta, ki jo nariše računalnik, zato ni ravna, temveč žagasta.

Lotimo se torej naloge! Piko za piko moramo pobarvati, drugo za drugo. Seveda moramo pobarvati začetno piko, naslednjo pa določiti nekje v smeri proti končni. Najlaže je določiti tisto, katere središče je najbližje točki, v kateri prava črta, to je zveznica začetne in končne točke, prečka sredino sosednjega stolpca.

Prestavimo se torej v sosednji stolpec, pogledjmo, kje ta stolpec prečka zveznica obeh točk. To nam povesta podobna trikotnika s slike 2. Iz enačbe podobnosti $14:24 = d : 1$ takoj izračunamo, koliko nad osnovnico trikotnika je presečišče, od tod pa dobimo njegovo višino v:

$$d = 14/24$$

$$v = 14 + d$$

Kako določimo presečišču najbližjo piko? Višino presečišča zaokrožimo, poiščemo ji najbližje celo število. V gornjem primeru moramo torej pobarvati piko z višino 15. Povsem podobno izračunamo presečišče zveznice z naslednjim stolpcem. Sami premislite, kako bi potrdili, da je vsako naslednje presečišče za d višje! Označimo z i, j koordinati pike, ki jo pobarvamo. Zgornji razmislek potem uredimo v tale postopek:

- (1) postavi se v začetno piko: $i := 23, j := 14$,
postavi $v := 14$ (saj je v tem stolpcu presečišče kar prva pika),
izračunaj $d := 14/24$;
- (2) pobarvaj trenutno piko,
če je to zadnja pika, končaj;
- (3) izračunaj višino naslednjega presečišča: $v := v + d$,
izračunaj koordinati nove pike: $i := i + 1, j :=$ zaokrožen v ,
pojdi na (2);

Program v basicu hočete? Samo en oreh še stremo, pa ga zapišemo: naučiti se moramo zaokroževati. Kot vemo, zaokrožimo decimalno število, ki leži med celima številoma k in $k + 1$, na k , če je manjše od $k+1/2$, sicer pa na $k + 1$. Naj nam zato c označuje razliko med višino presečišča in prvim za pol zmanjšanim celim številom pod njim. Višina začetne pike je celo število, zato je začetna

vrednost c kar $1/2$. Ko se preselimo v sosednjo piko, se c poveča za d . Pri tem lahko seveda spleza nad 1 , tedaj pač, ko v zraste čez naslednje, za pol zvečano celo število. V tem primeru moramo višino pobarvane pike zvečati za 1 , zato pa c odštejemo 1 . Ozrivo se nazaj, pa lahko hitro ugotovimo, da v sploh ne potrebujemo več, zadostujeta nam j in c .

Pa zapišimo sedaj program! Na primer takole (zaradi čitljivosti smo ga zapisali malo po svoje):

```
100 LET i = 23:
    LET J = 14:
    LET c = 1/2:
    LET d = (47-23)/(28-14)
```

Nasmehnili ste se: davno ste se že naučili, da je $47-23 = 24$. Res je, vendar nam zapis pokaže pot v posplošitev programa. Saj hočemo s črto zvezati poljubni par točk, kajne? Denimo začetno točko s koordinatama iz, jz s končno točko, katere koordinati zapišimo z ik, jk . Takoj popravimo ta korak programa, pa je program takle:

```
100 INPUT "Začetna točka", iz, jz:
    INPUT "Končna točka", ik, jk:
200 LET i = iz:
    LET j = jz:
    LET c = 1/2:
    LET d = (jk-jz)/(ik-iz)
```

Nadaljujmo: pobarvati je treba piko in preveriti, če smo na koncu črte:

```
300 PLOT i, j:
    IF i = ik AND j = jk THEN STOP
```

nato pa določiti koordinati nove pike:

```
400 LET i = i+1: LET c = c + d:
    IF c > 1 THEN
        LET j = j + 1: LET c = c - 1
500 GO TO 300
```

Ste program natipkali? Poigramo se malo z njim! Poženimo ga in vtipkajmo gornje podatke: 23, 14 pa 47 in 28. Pri vtipkavanju števila ločite med sabo s pritiskom tipke ENTER. Program lepo nariše črto in konča. Krasno, si mislimo, čas je, da ga posnamemo na kaseto.

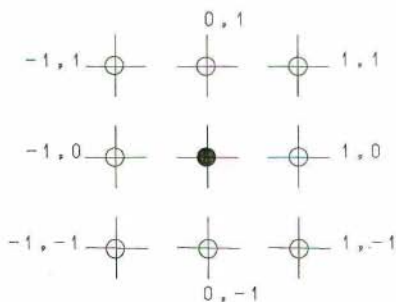
Pa malo dodatnega igranja – "testiranja" – nič ne škoduje. Kar poskusimo! Poženimo program ponovno, pri tem pa zamenjajmo začetno in končno točko! Ste videli? Program je odpovedal. Poskusimo še enkrat, tokrat zamenjajmo le stolpce z vrsticami, torej: 14, 23, 28, 47. Tudi tokrat program povsem zgreši končno točko. Karkoli poskušamo, program riše povsem pravilno le v desno navzgor, in še to le, ko črta ne raste prehitro, ko je razlika vrstičnih števil manjša od razlike stolpčnih števil.

Dopolnimo program, naučimo ga risati še v ostalih smereh! Zato si še enkrat oglejmo, kako izberemo novo piko za barvanje. O tem odloča ostanek c : če je manjši od 1, se pomaknemo vodoravno v desno, sicer pa poševno v desno navzgor. Izbiramo torej le med prvo in drugo sosedo pobarvane pike (slika 3). Program moramo torej popraviti tako, da bo znal izbirati še med preostalimi sosedami. Med katerimi? Odgovor razberemo iz slike: pri vsaki od črt bomo sosedo izbrali med tistima, med katerima pelje narisana črta. Do ene od obeh pridemo pošev, do druge pa vodoravno ali navpično. Poglejmo, kako bi ta koraka izračunali! Označimo poševni korak s pi , pj , pa takoj uganemo:

pi = predznak (ik–iz)

pj = predznak (jk–jz)

Funkcije predznak se seveda ne ustrašimo, tudi v basicu smo jo že srečali: SGN ji je bilo ime. Le ponovimo, da je predznak pozitivnega števila enak 1, negativnega -1 , pri 0 pa je enak 0. Lotimo se sedaj še drugega koraka, označimo ga z di , dj ! Korak bo vodoraven ali navpičen; zato bo eden od di ali dj enak 0, dru-



Slika 3. Sosedne pike

gi pa tak kot pi oziroma pj. Kateri? Tudi tu se ne zamudimo kaj dlje: če je velikost razlike vrstic večja od velikosti razlike stolpcev začetne in končne točke, bo treba korakati navpik, sicer vodoravno. Velikost števila nam v basicu izračuna funkcija ABS (absolutna vrednost).

Sedaj premislimo še, kako se bomo odločili med obema korakoma. Spet bo sodilo c. Kadar bomo izbirali med vodoravnim in poševnim korakom, nam bo povedal, kdaj moramo spremeniti vrstično številko; ko se bomo odločali med poševnim in navpičnim, pa bo odločil, kdaj spremenimo stolpčno številko pike, ki jo bomo pobarvali. Pomen c bo v obeh primerih podoben kot zgoraj: ko bo njegova vrednost zrasla nad ena, bomo izbrali poševni, sicer pa vodoravni oziroma navpični korak. Od pike do pike bo c spet naraščal s korakom d. Dopolnimo sedaj s temi ugotovitvami postopek za risanje:

- (1) preberi koordinate začetne in končne točke: iz, jz, ik, jk;
- (2) izberi sosedi in določi koraka do njiju:
pi := predznak (ik-iz), pj := predznak (jk-jz),
di := pi, dj := pj,
če je velikost (ik-iz) večja od velikosti (jk-jz)
potem postavi dj := 0 sicer pa di := 0;
- (3) izračunaj korak sodila c:
c := 1/2,
če je velikost (ik-iz) večja od velikosti (jk-jz)
potem postavi d := velikost ((jk-jz)/(ik-iz))
sicer pa d := velikost ((ik-iz)/(jk-jz));
- (4) postavi se v začetno piko: i := iz, j := jz,
- (5) pobarvaj trenutno piko,
če je to zadnja pika, končaj;
- (6) izračunaj vrednosti sodila v sosednjem stolpcu: c := c + d;
če je c večji kot 1
potem izberi poševni korak: i := i + pi, j := j + pj ter
zmanjšaj c: c := c - 1
sicer pa vodoravni oziroma navpični korak: i := i + di, j := j + pj,
pojdi na (5);

Na koncu zapišimo še program. Dopolnili smo ga toliko, da smo se znebili decimalnih števil: vsa števila, s katerimi računamo, so sedaj cela. Kako? Vrednost d in c smo pomnožili z imenovalcem d in z 2, temu smo ustrezno spremenili tudi vrednost 1 v primerjavi iz koraka (6).

```

100 INPUT "Zacetna tocka " /
    "stolpec = ";iz,
    "vrstica = ";jz
110 INPUT "Koncna tocka " /
    "stolpec = ";ik,
    "vrstica = ";jk
200 LET ri=ik-iz:
    LET rj=jk-jz:
    LET pi=SGN ri: LET di=pi:
    LET pj=SGN rj: LET dj=pj:
    LET ri=ABS ri:
    LET rj=ABS rj
210 IF ri<rj THEN
    LET di=0: LET d=2*ri:
    LET c=rj: LET m=2*rj:
    GO TO 240
220 REM ELSE
230 LET dj=0: LET d=2*rj:
    LET c=ri: LET m=2*ri
240 REM
300 LET i=iz: LET j=jz
400 REM LOOP
410 PLOT i,j:
    IF i=ik AND j=jk THEN
        GO TO 700
500 LET c=c+d
510 IF c<m THEN
    LET i=i+di: LET j=j+dj:
    GO TO 540
520 REM ELSE
530 LET i=i+pi: LET j=j+pj:
    LET c=c-m
540 REM
600 GO TO 400
700 STOP

```

Za primer uporabe opisanega postopka pa naj poskrbi tale naloga:

Napiši program, ki prebere koordinate treh točk in nariše pobarvan trikotnik z oglišči v podanih točkah!

Kdaj drugič pa si bomo ogledali, kako računalnik riše krožnice in elipse.