

## Kezelési útmutató a PFgeo-PC-E500S programhoz

A program alkalmas alappontsűrítési, részletmérési és numerikus területszámítási feladatok számítására. A beírt vagy számított pontokat letárolja, és azok a további számításokhoz felhasználhatók. A tárolt pontokra pontszámukkal hivatkozhatunk. A pontszámok az 1 -től 9999999998 -ig terjedő zárt intervallumba eső pozitív valós számok lehetnek. A tárolható pontok száma 249 db. Ha a tárolt pontok száma meghaladja a 239-et, a program minden újabb pont letárolásakor kiírja a szabad ponthelyek számát. Ha az újonnan számított pont már a 249-edik lesz, a program üzenetet ír a kijelzőre, de a számítások a sokszögvonala számítás és a területosztás kivételével tovább folytathatók, csak a továbbiakban számított pontok nem tárolódnak.

A program a pontokat mindig pontszámaik növekvő sorrendjében tárolja, a keresések meggyorsítása és a rendezett listák készítésének biztosítása érdekében. Minden újonnan beírt vagy számított pontot ennek megfelelően helyez el a már tárolt pontok között. Ezért célszerű a pontokat sorszámuk növekvő sorrendjében beírni, vagy számítani, mert a rendezés időigénye így minimális. Az alappontokat célszerű tizedesponttal tagolt formában letárolni (pld. 79-1063-at 79.1063-ként, mert így a magasabb pontszámú új pontok berendezésekor ezeket nem kell a programnak elmozdítania az adatbázisban.

Lehetőség van arra is, hogy a számításokat addig még le nem tárolt pontok felhasználásával végezzük és megszabhatja a felhasználó, hogy az ilyen pontok letárolódjanak-e vagy sem. Az újonnan számított pontok esetében is meghatározható, hogy azok letárolódjanak-e, vagy csak kijelze a program a számított koordinátákat.

A tárolt pontok a memória erre a célra fenntartott részére, vagy IBM PC -re elmenthetők vagy onnan beolvashatók.

A program fel van készítve a **SHARP CE-126P** típusú nyomtató kezelésére.

Ha a gép bekapcsolásakor üzemkés a nyomtató, akkor a számításokkal párhuzamosan nyomtatja a bemenő adatokat és számítási eredményeket. Ha a ilyenkor a nyomtatót kikapcsoljuk, vagy más okból üzemképtelenné válik, a kijelzőn a **Printer error in .....** üzenet jelenik meg. Ilyenkor a gépet ki-be kell kapcsolni és ha a nyomtató továbbra sem üzemkés, a gép figyelmen kívül hagyva a nyomtatóra vonatkozó parancsokat zavartalanul dolgozik tovább.

Ha printer nélkül használtuk a programot és használatba kívánjuk venni a printert, akkor a **BREAK (ON)** billentyűvel meg kell szakítani a program futását, a gépet **OFF**-al kikapcsolni. A nyomtató üzembe helyezése után a gép bekapcsolható és a program a továbbiakban már használni fogja a nyomtatót.

Mivel a program és az adatok a memóriát majdnem teljesen lefoglalják, tartózkodjunk újabb programsorok bevitelétől, és parancs üzemmódban ne használjunk kétbetűs azonosítójú változókat, mert ez a tárolt adatok elvesztését, vagy a program hibás működését okozhatja. Ne használjuk a standard szövegváltozókat se !

A standard numerikus változókat ( A-tól Z-ig ) nyugodtan használhatjuk !

**SZINTÉN TARTÓZKODJUNK A BEÉPÍTETT SZOFTVEREK HASZNÁLATBA VÉTELÉTŐL, MERT AZOK A RAM-BAN TÁROLT PROGRAMOT ÉS ADATOKAT KITÖRÖLHETIK.**

A program betöltés utáni inicializálása **RUN ENTER** utasítással történik.

Ezt használhatjuk a tárolt adatok teljes törlésére is. (Tehát ne használjuk addig, amíg a tárolt adatainkat nem akarjuk törölni !)

A gép nem rendelkezik a más SHARP kisgépeknél megszokott DEF billentyűvel ezért a program a **PF1** billentyűvel indítható újra, ha a tárolt adatokat meg kívánjuk őrizni. A PF1 billentyű a program főmenüjét aktivizálja, ahonnan azután az összes számítási, vagy egyéb funkció közvetlenül, vagy újabb menükön keresztül elérhető. A menük három vagy több karakterből álló emlékeztető kóddal jelölik a belőlük elérhető funkciókat. A kívánt funkció mindig az emlékeztető kódjában szereplő nagybetűk beírásával és az **ENTER** billentyű lenyomásával aktivizálható. Az egyes funkciók lefutásuk után automatikusan a megfelelő menühöz térnek vissza, ha van egyértelmű befejezési pontjuk. Ha nincs, vagy befejezés előtt akarunk visszalépni, akkor bármelyik pontszám bekérésre szolgáló inputnál **M ENTER** válasszal érhetjük ezt el. Az egyéb visszalépési lehetőségek az egyes funkciók tárgyalásában vannak részletezve.

A **PF1** lenyomása után megjelenik a program főmenüje az említett emlékeztető kódokkal.  
Jelentésük:

**Pol = Poláris pontok koordinátaszámítása az álláspont súlyozott tájékozáásával.**

Az álláspont tájékozáására a tájékozás funkcionál leírtak érvényesek. A részletpontok számítása folyamatos, a főmenühöz való **visszatéréshez az ALLASPONT ? , TAJ. PONT ? , PONT ?** Inputra **M ENTER** -rel kell válaszolni.

**Der = Derékszögű beméréssel meghatározott pontok számítása.**

A program bekéri a kezdő és végpont számát, kiírja a számított távolságot, és kéri a mért távolság beírását. Ha itt csak az **ENTER** billentyű lenyomásával válaszolunk, akkor a mért távolság felveszi a számított távolság értékét. Ez érvényes a vonalpont számításnál és a derékszögű kitűzési méretek számításánál is.

A program kijelzi a mért távolságot és az alapvonal hibáját, majd rátér a részletpontok számítására. Megszakítás az előbbiek szerint, bármelyik pontnál.

**Von = Vonalpontok számítása**

Működése megegyezik a derékszögű beméréssel.

**Met = Végpontjaikkal adott egyenesszakaszok metszéspontjainak számítása.**

A program úgy működik, hogy az elsőnek megadott szakasszal (alapvonal) több másik szakasz metszése is számítható, a funkció egyszeri meghívása alkalmával.

Új alapvonalra való áttérésnél a menü keresztül újra meg kell hívni a funkciót.

A funkció alkalmas koordináta tengelyekkel párhuzamos szakaszok metszésének számítására is. (Szelvénykeret átmetszés) Megszakítás az előbbiek szerint, bármelyik pontnál.

**SZm = Szelvénykeret átmetszés számítása.**

Belépés után először meg kell adni, hogy **Y** vagy **X** lesz az adott koordináta, majd meg kell adni annak értékét.

Ezután megkezdi a pontpárok bekérését és a metszéspontok kijelzését.

**lvm = Ívmetszés számítása**

Az előírás sorrendje az új pontról nézve, az óramutató járásával ellenkező értelmű. (Jobb->Bal) Egyszeri meghívással több új pont is számítható. Megszakítás az előbbiek szerint.

**Kit = Derékszögű kitűzési méretek számítása.**

Működése hasonló a derékszögű bemérésnél leírtakhoz.

**IRs = Irányszög és távolság számítása.**

Először a pólus számát kéri, majd az összeszámítandó pontét. Kiírja az irányszöget és távolságot, majd újra a pólust kéri. Ha itt csak **ENTER** -rel válaszolunk a pólus nem változik. Megszakítás az előbbiek szerint.

**Tav = Távolságszámítás**

Működése megegyezik az előbbivel de csak a távolságot írja ki.

**TAJ = Iránymérések tájékozása.**

Súlyozott tájékozást számít, max. 29 tájékozó pontra. Az álláspont bekérése után bekéri hogy akarjuk -e redukálni a mért távolságokat, ha **I** -vel válaszolunk akkor a mért távolságokat vízszintesre, tengerszintre és vetületre redukálja. A tengerszintre redukáláshoz bekéri a munkaterület átlagos magasságát. Ha erre csak **ENTER** -rel válaszolunk a magasság értéke 100 m lesz.

A vízszintesre redukáláshoz a mért távolság beadása után bekéri a zenitszöget. Ha erre csak az **ENTER** -rel válaszolunk, az aktuális távolság semmilyen redukciót nem kap. Ha csak a vízszintesre redukálást akarjuk kijelteni adjunk meg 90 fokos zenitszöget. Ez pontonként lehetséges. Ha a tengerszintre redukálást akarjuk kihagyni, akkor az álláspont megadása után adjunk meg 0 értékű magasságot. A vetületre redukálást a **Vet** funkcióval a főmenüből tilthatjuk le.

Ha az adott tájékozó pontra nem mértünk távolságot, a **MERT TAV ?** inputra csak az **ENTER** -t kell lenyomni. Az összes tájékozó adat bevitele után a **TAJ. PONT ?** inputra **Z ENTER** -rel válaszolva kezdeményezhetjük a középtájékozási szög kiszámítását és a tájékozási eltérések kimutatását. Megszakítható mind az **ALLASP ?** mind a **TAJ. PONT ?**

inputnál **M ENTER** -rel.

Ezután a program olyan pontok számait, a rájuk menő irányokat, és az esetleg mért távolságokat kéri be, amelyekre tájékozott irányértéket kívánunk számíttatni, vagy a távolságot szeretnénk redukáltatni. Ha nincs ilyen akkor **M ENTER** -rel léphetünk ki. Az itt leírtak érvényesek sokszögvonal végpontjának, előmetszés álláspontjának, vagy poláris számításnak a tájékozására is.

### **Sok = Sokszögelés**

Egy almenü jelenik meg: **2\* 1\* B S** -> A menüpontok sorra kétszer tájékozott, egyszer tájékozott, beillesztett, ill. szabad sokszög vonal számítását jelentik.

A választott típusnak megfelelően a program bekéri a kezdőpontot zárópontot, ha kell a tájékozó adataikat, majd a sokszögpontok adatait. Ha beírtuk az összes adatot a szabad sokszögvonal számításának kivételével a **SOKSZ. P ?** inputra adott **V ENTER** válasszal indítható a számítás. Szabad vonalnál meg kell adni az utolsó sokszögpont számát is, és az **IRANY H. ?** inputra kell **V ENTER**-rel válaszolni. Az **M ENTER** válasz mindkét esetben a főmenühez való visszatérést eredményezi.

A program kiírja a kimutatható eltéréseket és a sokszögpontok, valamint a zárópont számított koordinátáit, majd visszatér a menühez.

A sokszögpontokat a program már az adatbeírás fázisában letárolja. Ha tehát a program futása valamilyen okból megszakad a számítás befejezése előtt, ( A 99.99 m -nél nagyobb vetületű hossz zárást pl. nem tudja kiírni ) akkor a sokszögpontok törléséről külön kell gondoskodnunk.

Ha a kezdőpont bevitele után a redukálásra vonatkozó kérdésre I-vel válaszoltunk akkor a sokszögoldalak is redukálhatók, a tájékozásnál leírtak szerint.

A tájékozó pontok száma szabad és egyszer tájékozott vonalnál 29, kétszer tájékozottnál a kezdőponton 29 a záróponton 28 lehet. A számítható új pontok száma szabad vonalnál 29, egyébként 28 lehet.

### **Elo = Előmetszés, a meghatározó pontok tájékozásával.**

Lásd a tájékozás funkcionál leírtakat.

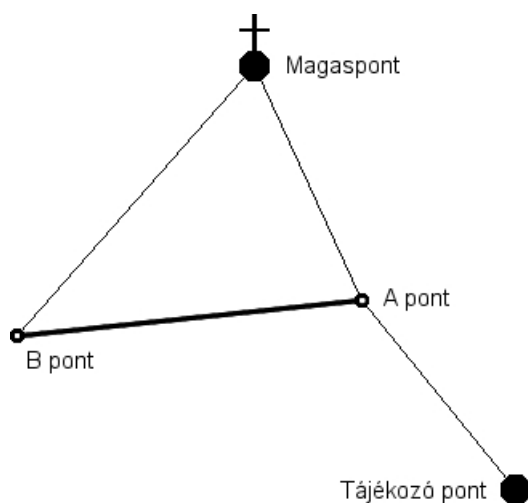
Az új pont koordinátáinak kiszámítása és kijelzése után a főmenühez tér vissza.

### **Hat = Hátrametszés számítása.**

Az új pont koordinátáinak kiszámítása és kijelzése után a főmenühez tér vissza.

### **MAp = Magaspont levezetés számítása**

Az alapvonal mindkét végpontjának koordinátáit számítja. Az alkalmazott jelöléseket lásd az alábbi ábrán:



### **KIE = Egyponthoz tartozó kiegyenlítés számítása.**

A kiegyenlítendő pont előzetes koordinátáiból, a mért belső irányok, távolságok és számított külső irányok (külső tájékozott irányértékek) ismeretében, a Gauss-féle legkissebb négyzetek módszerével meghatározza a koordináták legvalószínűbb értékét.

A külső irányokat a kiegyenlítendő pontról a meghatározó pontra mutató irányként kell beadni.

A kiegyenlítéshez minimum három meghatározó adatra (irány és/vagy távolság) van szükség.

Az irányokat hosszukkal egyenes arányban, a távolságokat hosszukkal fordított arányban súlyozza a program. A felhasználó csak a meghatározó elemek ismétlésével befolyásolhatja a meghatározó elemek kiegyenlítésre gyakorolt hatását. Például ha egy pontra csak belső irányt mértünk akkor úgy növelhetjük a súlyát a meghatározásban ha az irányt külső irányként is bevisszük. Másik megoldás az lehet ha a nagyobb súllyal szerepeltetni kívánt pontot többször szerepeltetjük a meghatározó pontok között.

Meghívása után először a pontszámot kéri amellyel a kiegyenlítendő koordinátákat letároltuk, majd rátér a meghatározó pontok pontszámának és a róluk vagy rájuk mért vagy számított meghatározó adatok bekérésére. Amelyik meghatározó adat nem létezik, annál csak az **ENTER** billentyűt kell lenyomni.

Ha vannak mért belső irányok, akkor az első meghatározó pontnak olyan pontot kell választani amelyen van belső irány.

A meghatározó pontok száma legfeljebb 8 lehet.

A számítás csak az utolsó meghatározó pont összes adatának bevitele után visszatérő **MEGH. PONT ?** inputról indítható **K ENTER**-rel. Ha vannak belső irányok, a program kijelzi az előzetes középtájékozási szöveget, majd pontonként a meghatározó elemek eltéréseit mutatja ki. Ezután kezdi meg a kiegyenlítés számítását.

Mivel a tájékozási ismeretlen kijelzése iterációval történik, ez a program fázis a meghatározó pontok számától függően hosszú ideig tarthat. A program minden iterációs ciklus végén kijelzi a koordinátaváltozásokat. Ha azok összege kisebb 0.005 m -nél, befejezi a kiegyenlítést.

A kiegyenlítés befejezése után a program kijelzi a súlyegység és a kiegyenlített koordináták középhibáit, a koordinátaváltozásokat és a kiegyenlített koordinátákat. Ezután kimutatja a meghatározó elemeknek a kiegyenlített koordinátákkal számítható eltéréseit, majd a menühez tér vissza.

### **Aff = Affin transzformáció**

Először a II. rendszer nevét kéri be. Ez legyen EO, St., HER vagy más néhány karakterből álló azonosító, de akár el is hagyható, ha csak az **ENTER** -t nyomjuk le.

Ezután megkérdezi a program, hogy előzőleg már számított és letárolt, a billentyűzetről most beírandó, vagy most számítandó állandókkal kívánunk dolgozni. A lehetséges válasz **T**, **B**, vagy **S** karakter és az **ENTER** lenyomása. (**M** -re a menühez lép)

-Ha **S ENTER**-rel válaszoltunk, bekéri a közös pontok számait és II. rendszer béli koordinátáit. A közös pontok száma max. 29 db. Mivel az affin transzformáció számításához legalább három pont kell, három pontra viszont mindig ellentmondásmentes, ezért használjunk legalább négy pontot.

A közös pontok új rendszerbeli koordinátáinak megadása után a **PONT ?** inputra **A ENTER** -rel válaszolva indíthatjuk az állandók számítását. A program kiírja a számított állandókat és meg kérdezi kívánja-e a felhasználó letárolni az állandókat I válasz esetén az állandók a memória erre a célra kijelölt területén

letárolódnak és a későbbiek során felhasználhatók mindaddig amíg újabb számított vagy beírt állandók letárolásával felül nem írjuk őket, vagy a programot **RUN ENTER** paranccsal alapállapotba nem hozzuk. N válasz esetén a számított vagy beírt állandók a menühez való visszatéréskor elvesznek.

Ezután a program áttér az átszámítandó pontok bekérésére.

-Ha **B ENTER**-rel válaszoltunk, a program sorra bekéri az a,b,c,d transzformációs állandókat és a hozzájuk tartozó I. és II. rendszer béli közepes koordinátákat, majd kijelzi azokat és az előzőek szerint felajánlja a letárolás lehetőségét, majd áttér az átszámítandó pontok bekérésére.

-Ha **T ENTER** -rel válaszoltunk, a program kijelzi a tárolt állandókat és áttér az átszámítandó pontok bekérésére.

Ha nem új állandók számítását választottuk, a program lehetőséget biztosít az átszámítás irányának megválasztására is, tehát ugyanazokkal az állandókkal mindkét irányban tudunk transzformálni.

Ha a program \* **TAROL** \* állapotban van, az átszámított pontok eredeti koordinátáit az átszámított koordináták felülírják, a szándékolatlan felülírás elkerülése érdekében a program az átszámítás megkezdése előtt automatikusan \* **NEM TAROL** \* állapotra áll be, **FIGYELEM** ! Az előbbiekből az is következik, hogy ha \* **TAROL** \* állapotra állítjuk vissza a programot és a transzformálandó pontokat átszámítás közben, negatív pontszám segítségével adjuk meg a programnak, akkor is a transzformált koordináták kerülnek be az adatbázisba.

#### **Koz = Irányok és távolságok központosítása.**

A központosítás történhet külpontosan, vagy központosan mért, vagy a tárolt koordinátákból számított távolság felhasználásával. A program bekéri mind a külpontosan, mind a központosan mért távolságot. Amelyiket nem mértük, arra csak az **ENTER** billentyűvel válaszoljunk. Ha mind a központ, mind az irányzott pont tárolva van és nem mértünk semmilyen távot, akkor a program a számított távolságot használja. Ha az sem létezik, hibajelzés után rátér a következő pont számítására. Megszakítás az előbbiek szerint.

#### **BEi = Pontok számának és koordinátáinak letárolása.**

A funkció sorra kéri be a letárolandó pontok pontszámait és koordinátáit. A pontszámot szükség esetén tizedesponntal tagolhatjuk, pld.: a 123/1 pontot célszerűen 123.1 alakban, a 79-1063 pontot 79.1063 alakban adhatjuk be. A koordináták max. hat egész jegyet tartalmazhatnak. A tizedesek száma nincs korlátozva, de azokat a program azonnal két tizedesre kerekíti, figyelembe véve a páros fele kerekítés szabályát is. A főmenühöz a **PONT ?** inputra adott M válasszal léphetünk vissza.

#### **Kli = Tárolt pontok pontszámainak és koordinátáinak kijelzése.**

Meghívása után a program megkérdezi, hogy csak a pontszámokat kívánjuk kijeljeztetni, vagy a koordinátákat is. A kérdésre **I** vagy **N** karakterrel és az **ENTER** billentyűvel kell válaszolni. Alapértelmezett az **I** válasz, tehát ha csak az **ENTERT** nyomjuk meg, akkor a koordináták is kijelzésre kerülnek. Ezután az első listázandó pont számát kéri a program. Itt a legalacsonyabb pontszámú pont az alapértelmezett.

A kijelzőn megjelenik a kívánt pont.  
A következő pont kijelzését a az előző kijelzését a billentyű lenyomásával kérhetjük. A megfelelő billentyű többszöri lenyomásával az adatbázis bármely pontját elérhetjük.  
A billentyű lenyomása után a kijelzőn a legalacsonyabb a lenyomása után a legmagasabb pontszámú pont jelenik meg, és a billentyűkkel onnét folytatható a listázás.  
Ha kb. két percig nem használjuk egyik billentyűt sem a program visszatér a főmenühöz. Erre azért van szükség, mert a pontkijelzés állapotában az automatikus kikapcsolás nem működik és az így felejtett gép elemei lemerülhetnek.  
Az **ENTER** billentyű hatására a program újra bekéri az első listázandó pontot, így az adatbázis tetszőleges pontjára ugorhatunk.  
Az **M** billentyű hatására a főmenü tér vissza.  
A **J** billentyűvel a ponttörlő funkciót hívhatjuk meg egy pontintervallum törlésére. A törlés elvégzése után az első listázandó pont bekérésével folytatódik a funkció.  
Ez a funkció csak a kijelzőn jeleníti meg a pontokat. A nyomtatás hoz használjuk a következő fejezetben leírt funkciót.

#### **Lis = Tárolt pontok listázása nyomtatóra.**

Bekéri az első és utolsó listázandó pont pontszámát, kilistázza a pontokat majd új pontintervallumot kér.

#### **TaR = A funkció célja, közölni a programmal, hogy a továbbiakban számított új pontokat kívánjuk-e tárolni, vagy sem.**

A funkció minden meghívás alkalmával az éppen aktuális állapot ellentettjét állítja be, tehát mintegy ki-be kapcsolja a letárolást. Egyes számítási funkciók önállóan is állítják a letárolást. A beírás, vagy a területosztás funkció meghívása automatikusan \* **TAROL** \* állapotba állítja a programot. Sokszögvon



számításokor a sokszögpontok adatainak bekérése előtt szintén a \* **TAROL** \* állapot lesz érvényes, az affin transzformáció használatakor pedig a transzformálandó pontok bekérése előtt a \* **NEM TAROL** \* állapot. Az ilyen automatikus beállításokat az affin transzformáció kivételével nem szabad felülbírálni a felhasználónak, mert akkor a program hibásan működik.

A funkció elérhető számítás közben is mindazon számítási eljárásokból amelyek folyamatosan több pont számítására képesek, (A sokszögvonallal számítás nem ilyen.) az újonnan számítandó pontok pontszámát bekérő inputra **T ENTER** választ adva. Lefutása után a számítás zavartalanul folytatható.

A funkció ugyanígy elérhető olyan funkciókból is amelyek nem koordinátaszámítási feladatokat oldanak meg, mint a derékszögű kitűzési méretek számítása, irányszög-távolság számítás, iránymérés tájékozása, területszámítás tárolt pontokkal. Ennek az a jelentősége, hogy így ezen funkcióknál is kötetlenül kihasználhatjuk a program által biztosított azon lehetőséget, hogy egy az adott számítási funkció szempontjából adott pontnak minősülő, de addig még le nem tárolt pontot számítás közben is megadhatunk úgy, hogy pontszámát negatív előjellel adjuk be. Ilyenkor a program bekéri a pont koordinátáit, a \* **TAROL** \* vagy \* **NEM TAROL** \* állapotól függően véglegesen, vagy ideiglenesen letárolja a pontot, majd folytatja a számítást a következő lépéssel. E két lehetőség ésszerű kombinálásával, a felhasználó szabadon dönthet arról, hogy mikor adja meg a használni kívánt pontok adatait és hogy, le kívánja-e tárolni azokat, kivéve azokat az eseteket amikor a program önállóan állítja a tárolási állapotot.

A negatív pontszámmal történő koordináta megadás csak a beírt adatokból történő terület számítás funkcióban nem használható.

Ha a tájékozási funkciót (Poláris pontok álláspontjánál is!) úgy használjuk, hogy a tárolás \* **NEM TAROL** \* állapotú, akkor a negatív pontszámmal, számítás közben beadott pontok száma max. 5 db. Lehet a számításban.

Ha sokszögvonallal végpontjainak megadása és tájékozása történik az előbbi módon, akkor a kezdőpont max. 5 db. a zárópont max. 4 db. pontot adhatunk meg számítás közben negatív pontszámmal.

Ugyanez érvényes az előmetszés két álláspontjának tájékozására is.

\* **TAROL** \* állapotban, a számítás közben megadható pontok számát csak a szabad pontok száma (max. 249) korlátozza.

Ha egy újonnan számított, vagy negatív pontszám megadása után bevitt pont nem kerül végleges letárolásra koordinátáinak kijelzésekor a program hangjelzéssel figyelmezteti erre a felhasználót.

#### **VEt = Vetületek menüje: Eov St. Hen. \* ->**

A menüre **E, S, H**, \* betűvel válaszolva adhatjuk meg, hogy a továbbiakban a vetületi hosszredukciókat **EOV**, Stereo vagy Henger vetületben kell számítani, vagy nem kérünk vetületre redukálást.

#### **Ren = Pontok rendjének megadása a tájékozási eltérések kimutatásához.**

A **V.rendu Soksz.** -> menü jelenik meg. **V** válasz esetén a program az iránymérés tájékozási funkcióban és az ezt használó más funkciókban a tájékozási eltérések kimutatásakor kijelzi az ötödrendű háromszögelésnél az adott irányra megengedett hibát is, **S** válasz esetén pedig csak az eltéréseket mutatja ki.

#### **Num = A numerikus területszámítás menüje jelenik meg.**

A számítandó területnek a **0 m<sup>2</sup> <= T < 10000000000 m<sup>2</sup>** intervallumba kell esnie. A menüpontok jelentése:

#### **Nul = Az eddig számított területek összegének nullázása.**

#### **Tar = Területszámítás az adatbázisban tárolt, vagy negatív pontszám megadása után beírt pontokból.**

A funkció sorra kéri az elhatároló pontok számait. A kezdőpont újra beadása zárja le a számítást. Kíírja a számított területet és az addig számított területek

összegét, majd kéri a következő számítandó terület pontjait.

A numerikus területszámítás menüjéhez a **PONT ?** inputra adott **N** válasszal, a főmenühöz **M** válasszal léphetünk.

**J** válasza a ponttörlő rutin indul el és lefutása után a területszámítás folytatható.

#### **Bei = Területszámítás a billentyűzetről bevitt koordinátákból.**

A funkció sorra kéri az elhatároló **Y** és **X** koordinátákat. A kezdő koordinátapár újból beadása zárja le a számítást.

A menükhöz az **Y ?** vagy **X ?** inputra adott **N** vagy **M** válasszal juthatunk vissza.

#### **Jav = A számított területek összegének korrigálása.**

Kijelzi az összterületet, és bekéri a javítás mértékét. Ez maximum 10 egész és öt törtjegyet tartalmazhat. A javítás elvégzése után új adatot kér. A menükhöz **N** vagy **M** válasszal juthatunk.

#### **Men = Visszalépés a főmenühöz.**

#### **Hle = Háromszögek területének számítása Hieron képletével.**

Bekéri a három oldal hosszát és kijelzi a területet 7 tizedes, vagy összesen 10 számjegy pontossággal, utána újabb oldalhosszakat kér.

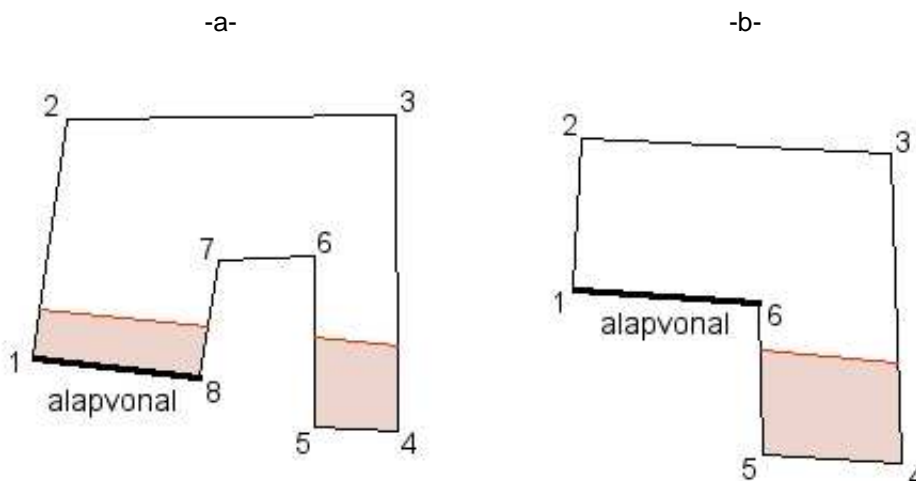
A menühöz bármely oldalnál megadott **M** válasszal térhetünk vissza.

A funkció nem használja a nyomtatót.

#### **TEo = Poligon felosztása valamely oldalával párhuzamos osztóvonalakkal, adott területű részpoligonokra.**

A poligon töréspontjainak és az osztópontoknak az együttes száma maximum 30 db. lehet. Ha helyi koordináta-rendszerben dolgozunk, ügyeljünk arra, hogy a számításba bevont összes pont ugyanabba a síknegyedbe essen. Helyi rendszer alkalmazásakor egyébként is célszerű a rendszert úgy kialakítani, hogy a munkaterület az első síknegyedbe essen, mert ilyenkor nem kell a koordináták előjelével foglalkoznunk.

Nem kezel helyesen olyan eseteket amikor a számítandó részterület több részből állna ( a. ábra ) vagy a számítandó részterület osztóvonal a kezdőoldalnak a kiosztás irányával ellentétes oldalára esne ( b. ábra )



Ilyen esetekben a poligont célszerű részekre bontva felosztani.

A funkció meghívása után először a poligon töréspontjait kell megadni. Az elsőként megadandó pont a kezdőoldalnak a kiosztás irányába nézve bal oldalra eső végpontja, az utolsó pedig eszerint a kezdőoldal másik végpontja. Az előírás végét az utolsó pont beadása utáni inputra adott **V** válasszal jelezzük a programnak.

A program kiszámítja és kijelzi a felosztandó poligon területét.

Mivel a felosztás nem az adatbázisban tárolt, hanem a kezdőoldalba eső +Y tengelyű rendszerbe átszámított kerekítetlen koordinátákkal történik ezért a felosztás során számított

területek kismértékben eltérhetnek az eredeti koordinátákból számítható területektől.

Ezután a program bekéri a részterületek számát.(Kiosztandó+viisszamaradó) A válasz egy +2 és +13 közé eső egész szám lehet.

A következő lépés a részterületek megadása. A beadott értéknek egy négyzetméter és a még kiosztható terület közé kell esni, egyébként a program megismétli a részterület bekérését. A beadott részterületnek a program csak az egész részt veszi figyelembe. Ha a részterület bekérésekor **M** választ adunk, a program a menühöz tér vissza.

A részterület megadása után a program fokozatos közelítéssel meghatározza a hozzá tartozó osztóvonal helyét és végpontjainak koordinátáit. A megadott részterületre való ráállítás hibája szokványos táblaméretetek esetén nem haladja meg a néhány század négyzetmétert.

Sajnos az osztópontok koordinátáinak végleges letárolásakor, a kerekítések miatt az osztóvonal és a csatlakozó határvonalak hosszától függő mértékben eltérhet a végleges numerikus terület a beadott értéktől.

A közelítés során a program kijelzi a közelítő területeket, a közelítés befejeztével pedig az addig kiosztott területek összegét is, majd bekéri a következő részterületet és megismétli a közelítési procedúrát.

Az utolsó előtti részterület kiszámítása után az **ENTER** lenyomását kéri, majd kiszámítja és kijelzi az utolsó részterületet.

Újra **ENTER**-t kér, a pontokat visszaszámítja az eredeti koordinátarendszerbe és kijelzi az osztópontok bevonásával számítható összterületet.

Ismét az **ENTER** lenyomását kéri és megkezdzi az osztópontok pontszámainak bekérését, koordinátáik és az általuk lemetsett részterületek végleges értékének és azok összegének kijelzését. Az osztópontok csak ebben a fázisban tárolódnak le az adatbázisban.

A pontszámok bekérésekor mindig kijelzi annak a poligonoldalnak a végpontjait, amelyre az adott osztópont esik. Az utolsó részterület kijelzése után az **ENTER** lenyomását kéri és visszatér a főmenühez.

A területosztás funkció lenullázza a numerikus területszámítás területösszegzését.

Bár a funkció csak a poligon valamely oldalával párhuzamos osztásra képes, a felosztandó poligon előzetes átalakításával merőleges, vagy egyéb iránnyal párhuzamos felosztásokat is megoldhatunk segítségével.

#### **Jav = Pontok törlése az adatbázisból.**

Először az első törlendő pont számát kéri be. Alapértelmezett érték a legkisebb pontszámú pont, 0 megadása a legnagyobb pontszámú pontot jelenti.

Ezután az utolsó törlendő pont számát kéri. Itt az elsőnek megadott pont az alapértelmezett érték, 0 megadása itt is a legnagyobb pontszámú pontot jelenti.

Az utolsó törlendő pont megadása után, ha a törlendő intervallum csak egy pontot tartalmaz azonnal végrehajtja a törlést, ha több pontot jelöltünk ki, előbb rákérdez, hogy valóban törölni kívánjuk-e azokat. I válasz esetén végrehajtja a törlést, N válasz esetén új intervallum határok bekérésére tér át.

A törlés végrehajtása után kijelzi a törölt intervallumot és új intervallum határokat kér.

A funkció a **Tar** funkciókódnál leírtakhoz hasonlóan számítás közben is elérhető, a megfelelő inputoknál megadott **J** válasszal. Ilyenkor azonban csak egy intervallum törölhető és utána a számítás zavartalanul folytatható. Így például hibásan számított pont azonnal törölhető és törlés után újra számítható.

A főmenühez az intervallum határok inputjaira adott **M** válasszal térhetünk vissza.

#### **Inf = Információk a program és az adatbázis állapotáról.**

Kijelzi az aktuális területet (E,S,H,\*), az aktuális pontrendűséget (V,S), a tárolás aktuális állapotát ( \* **TAROL** \* = 1 , \* **NEM TAROL** \* = 0 ). Kijelzi az adatbázisban tárolt pontok darabszámát és a legkisebb illetve legnagyobb tárolt pontszámot, valamint a **RAM** lemezen rendelkezésre álló szabad helyet.

Bármely billentyű lenyomására vagy kb. két perc várakozás után a főmenühez tér vissza.



**ATv = (Átváltások) Egy almenü jelenik meg, amelynek egyes pontjait aktivizálva a következő számításokat végezhetjük:**

**ME->öl = Meter -> öl átszámítás.**

Bekéri a méterben adott adatot és kijelzi öltre átszámított értékét, majd új adatot kér. A menühöz M válasszal juthatunk vissza.

**OL->mé = öl -> méter átszámítás az előző funkció mintájára.**

**M2->öl2 = Négyzetméter-> négyszögöl átszámítás az előzőek szerint.**

**O2->mé2 = Négyszögöl-> négyzetméter átszámítás az előzőek szerint.**

**HA->hold = Hektárban adott mennyiséget számít át hold + négyszögöl értékre.**

Az adatot tizedestört formában kell megadni:

2 ha 123 négyzetméter = 2.0123

2 ha 1167.56 négyzetméter = 2.116756

**HOld->ha = Hold.négyszögöl formában megadott adatot számít át hektár + négyzetméter értékre.**

Az adatot úgy kell megadni, mintha a négyszögölérték a hold érték tizedesrésze lenne:

2 hold 123 négyszögöl = 2.0123

2 hold 1167.56 négyszögöl = 2.116756

**SZög = Irányértékek, szögértékek előjeles összegzésére szolgál.**

A kapott adatot korrigálja, ha az negatív, vagy forgásszög lenne.

**IBM = Adatforgalom IBM kompatibilis PC -vel.**

Az IBM PC -n el kell indítani az adatok fogadását, vagy küldését végző programot (**PCE500.BAT**), meg kell adni a használni kívánt soros port számát, az adatátvitel irányát, a be vagy kiolvasandó adatok helyét (fájlnév), majd az IBM-PC által kijelzett utasítás szerint el kell indítani a tényleges adatátvitelt.

Az átvendő adatok körének meghatározására a programban nincs lehetőség, mert ez az IBM PC oldalán mindkét adatátviteli irány esetén könnyebben és áttekinthetőbben megoldható a legegyszerűbb szövegszerkesztővel is.

A beolvasandó adatokat tartalmazó fájlban a pontoknak pontszám szerint rendezetten kell szerepelniük. Ha a program rendezetlen adatokat kap, a nem megfelelő sorrend észlelésekor befejezi az adatok átvételét.

**RAM = Műveletek a RAM lemezzel.**

A memória egy része adattárolásra van fenntartva. Ezt nevezzük RAM lemeznek.

**MEM->ram = Az adatbázisban tárolt pontok elmentése a RAM lemezre.**

Bekéri az első és utolsó elmentendő pont számát, majd a fájl nevét, amelybe menteni szeretnénk az adatokat.

Ha a fájl még nem létezik, megkérdezi létrehozza-e, ha már létezik rákérdez, hogy felülírható-e az új adatokkal. Nemleges válasz (**N ENTER**) esetén új fájlnevet kér, egyébként elvégzi a mentést és a **RAM** műveletek menüjéhez tér vissza.

A **RAM** lemezen keletkező fájl mérete nem csak az elmentett pontok számától, hanem a tárolt adatok (pontszámok, koordináták) hosszától is függ. Ezért a fájl mérete nehezen lenne programból előre meghatározható. Előfordulhat tehát, hogy a mentés idő előtt megszakad a **\*NINCS ELEG HELY !\*** üzenettel.

**RAM->mem = A RAM lemezre korábban elmentett adatok beolvasása az adatbázisba.**

Megkérdezi a program, hogy a beolvasandó adatokkal felülírja-e az adatbázist az első tárolási helytől kezdve, vagy a beolvasott pontokat az adatbázishoz hozzáfűzze. Ezután az első és utolsó beolvasandó pontot kell megadni. Ha hozzáfűzést kértünk, akkor itt csak olyan pontszámok adhatók meg, amelyek nem szerepelnek az adatbázisban és az első beolvasandó pont számának nagyobbnak kell lennie az adatbázisban tárolt legnagyobb pontszámánál.

Ezután kell megadni a fájl nevét amelyből beolvasni kívánjuk az adatokat. Ha nem

létező fájl nevét adjuk meg a név bekérését megismétli. Ha a névre **M ENTER**-rel válaszolunk a **RAM** műveletek menüjéhez tér vissza.

Létező fájl nevének megadása után addig olvassa a fájl tartalmát amíg az elsőnek megadottal egyenlő, vagy annál nagyobb számú pontot nem talál. Ettől kezdve a beolvasott pontokat letárolja az adatbázisban mindaddig, amíg azok pontszáma kisebb, vagy egyenlő az utolsónak megadott pontszámhoz képest, vagy a fájl végére nem ér.

Ha nem talál a fájlban a feltételeknek megfelelő pontot akkor figyelmeztető üzenetet ír ki, egyébként kijelzi a beolvasott pontok darabszámát és a **RAM** műveletek menüjéhez tér vissza.

### **Ram->kij = A RAM lemezre korábban elmentett adatok listázása a kijelzőn, áttekintés végett.**

Bekéri a listázandó fájl nevét, majd megkezdí a benne található pontok adatainak listázását a kijelzőre. A listázás folyamatos, de a **SPACE** billentyű lenyomásával bármikor megállítható, majd az **ENTER** lenyomásával tovább folytatható. Ha az utolsó pont is a kijelzőre került, vagy lenyomtuk az **M** billentyűt a listázás megszakad, és az **ENTER** lenyomása után a **RAM** műveletek menüje tér vissza.

## **Egyéb tudnivalók**

A **RAM** lemezen található adatfájlok listáját megkaphatjuk, ha a **BREAK** és **PF2** billentyűket lenyomjuk.

A kijelzőn megjelenik a fájlok listája. A listában a billentyűkkel lehet előre hátra mozogni. A baloldalon látható nyíl az aktuális fájl jelöli. Ha a fájl törölni akarjuk, a **K** billentyűt kell lenyomnunk (Kill). A gép megerősítést kér (**OK?**) és ha **Y** (Yes) választ kap törli az aktuális fájl és a lista elejére áll.

A fájlok listázásából a **C\*CE** billentyű lenyomásával léphetünk ki és a **PF1** billentyűvel újra indíthatjuk a programot.

A program az áttekinthetőség érdekében a numerikus adatok többségét meghatározott formátummal írja a kijelzőre. A formátumok úgy vannak meghatározva, hogy azokban a gyakorlatban előforduló numerikus értékek elférjenek.

Hibás adat (pl. háromnál több egész jegyet tartalmazó irányérték, vagy hatnál több egész jegyet tartalmazó koordináta stb.) vagy a gyakorlatban előfordulhatót meghaladó nagyságú számított adat előfordulása esetén a program "**Can't print in specified format**" üzenet kijelzése mellett megszakadhat. Ilyenkor a programot újra kell indítani (**C.CE PF1**) és miután a hiba okát feltártuk (Hibás adott koordináta, hibás mérési adat bevitele stb.) a számítást meg kell ismételni. Az ilyen jellegű programmegszakadások előfordulását csak az összes bemenő adat megfelelő szempontok szerinti ellenőrzésével lehetne megakadályozni programból. Ez viszont a programkód méretét jelentősen növelné, ami a számítási, vagy pont tárolási lehetőségek csökkenésével járna. Ezért inkább a felhasználót kérem, legyen körültekintő az adatok bevitele során.

Ha a program egy adott (letárolt) pont megadását várja és a felhasználó az adatbázisban még nem szereplő pontszámot ad meg, a program hangjelzéssel és \* **NINCS** \* hibaüzenettel figyelmeztet, és **PONT UJRA !** szöveggel újra bekéri a pontszámot. Ezt mindaddig ismétli, amíg tárolt pont számát nem kapja, vagy negatív pontszám megadása után be nem adjuk a koordinátákat, vagy meg nem szakítjuk a funkciót a menühez visszatérve.

Ha a program egy letárolandó pont számára kér pontszámot és az adatbázisban már szereplő adatot kap, hangjelzéssel és \* **FOGLALT** \* hibaüzenettel figyelmeztet, majd **PONT UJRA !** szöveggel addig ismétli a pontszám bekérését amíg valóban új pontszámot nem kap, vagy vissza nem lépünk a menühez.

Ha a pontszámot bekérő input **PONT ?** szövegű, akkor ha csak az **ENTER**-rel válaszolunk rá automatikusan az előző ilyen inputnál beadott pontszámnál eggyel nagyobb pontszámot generál. Így például poláris részletpontok számításánál, ha a pontok számozása folyamatos és növekvő, elegendő csak az első pontnál beírni a pontszámot és a többi pontnál csak az **ENTER** -t kell lenyomni.

1994. 11. 21.

Papp Ferenc