

Hogyan programoztunk mi? Emlékszilánkok

Vértés János

1967. Ekkor kezdtem meg matematikai tanulmányaimat a Leningrádi Állami Egyetem matematika szakán, és ebben az évben jelentek meg a Szovjetunióban azok a rendeletek, amelyek – lényegében – elindították az ESZR programot. Ekkorra készült el a „RJAD” (Egységes Számítógéprendszer, ESZR) pilot projektje, és mivel ez az IBM 360 rendszerének másolatán alapult, minden egyetememen bevezették a kötelező számítástechnikai képzést, amelynek alapját – nem véletlenül – épp az IBM 360 jelentette. (A matematika szakon szerencsére nem, de mindenütt, ahol a tananyag távol állt az informatikától, még a tanulmányi időt is megnövelték egy szemeszterrel, hogy ne jöhessen ki a jövőben az egyetemekről olyan diplomás, aki nem rendelkezik számítástechnikai ismeretekkel.)

A hardware alapismeretek mellett nekünk az ALGOL 60 programozási nyelvet kezdték el tanítani, amit az első év utáni nyári gyakorlatomon, a Kun Béla téri SZÜV-ben (hol van mindez már: SZÜV, Kun Béla tér... pont ott, ahol Leningrád, Szovjetunió, Zsdanov egyetem...) már használni is tudtam. Lényegében itt készültek az első programcskáim a SZÜV-nél található GIER-re. A programsorokat papírra írhattam, ezeket lyukkártyára rögzítették, a gép ellenőrizte, kidobta a szintaktikai hibákat. Akkoriban egy programozó a gép közelébe sem mehetett, leadtuk a kódot, visszakaptuk a hibát, a kettő között ment – a feladatokat batch-be rendezve – a feldolgozás, a drága gépidővel nem mi rendelkezünk. Amikor a szintaktikai hibákat kijavítottuk, compilerrel gépi kódra fordították a programot, és akkor az vagy azt csinálta, amit kellett, vagy kereshettük a szemantikai hibát. Hogy miről szóltak azok a nyári gyakorlatos programok? Például kaptunk egy tekercs lyukszalagot, amit valamelyik főnök a nyugati útján elcsent valahonnan, és az volt a feladat, hogy ezt olvassuk be a géppel, és írassuk ki a tartalmát, majd próbáljuk megfejteni. Arra már nem emlékszem, hogy sikerült-e, csak arra, hogy egyszer félórával a munkaidő vége előtt elkéredzkedtem. Akkor kötöttem az első házasságom. (Mondjuk az nem a legjobban sikerült, túl fiatal voltam.)

Leningrád legelső számítógépe az Ural-1 volt, amit a „mi” mat-mech karunkon helyeztek üzembe még 1957-ben. Ez ugyan nemigen lett volna beilleszthető az IBM System 360 sorozatba, de a korszerűbb elmélethez a gyakorlatot mi ezen kezdtük el 1967-68-ban tanulni. Ekkor azonban már rohamosan kezdett bővülni a géppark, sorra érkeztek az új „korszerű”

komputerek: M-20, BESM-3M, M-222, Odra-1204, de ezek egyike sem tartozott a RJAD sorozatba. Az IBM 360-as sorozatra hajazó (többé-kevésbé) ESZR gépekből az 1972-ig tartó tanulmányaim alatt eggyel sem sikerült találkozni, az R10 elődjei (Mitra 15, Videoton 1010B) megvárták, hogy hazajöjjenek, és Székesfehérvárra költözzünk.

Engem ezek a – a byte-ok tartalmát villogó lámpasorokkal is megmutató – gépcsodák a Néva-parton még nem ejtettek rabul, a pszeudo-differenciál operátorok annál inkább. Professzorom, Szolomon Grigorjevics Michlin óvott az új divattól, a diplomavédést követően ezzel bocsátott útra az életbe: „Ne add el a lelked a számítástechnikának, maradj elméleti matematikus!”

Nos, hazatérve a még meleg matematikusi diplomámmal egy fél pillanatig valóban fontolóra vettem, hogy megmaradjak kedvenc operátoraimnál, de hamar kiderült, hogy erre minimális Magyarországon a kereslet, így a számítógéptermi operátorok környékén kellett körülnézni. Pusztán geopolitikai megfontolásokból (mindössze egy órányi vonat- és busz-távolságra van Budapesttől, mégsem Budapest) a SZÜV-ös nyári gyakorlat óta már meglévő feleségemmel Székesfehérvár mellett döntöttünk, itt a Videoton Számítástechnikai Gyára tárt károkkal és kiemelkedően magas (2400 Ft + 400 Ft albérleti hozzájárulás) fizetéssel várt, ráadásul az újságírás iránti vonzalmamat is tudtam kamatoztatni, a *Fejér megyei Hírlap* örömmel foglalkoztatott külső munkatársként.

A fehérvári Videotonnál Nagy Géza vezette a programozókat, így tőle kaptam az első komoly feladatomat, egy általános sorba-rendező programot kellett írni.

A kihívást az jelentette, hogy a 1010B memóriája korlátozott volt, tehát a rendezést két részre kellett bontani. Azt az adatmennyiséget, ami elfért a memóriában, sorba rendezés (sort) után ki kellett írni mágnesszalagra, majd a mágnesszalago(ko)n lévő rendezett stringeket össze kellett fésülni (merge) egyre hosszabb stringgé, egészen addig, amíg a teljes adattömeg egyetlen egy rendezett stringet nem alkotott. A Videoton 1010B háttérében általában négy mágnesszalagegység állt, ezért az első idea az volt, hogy a sort szakasznál két meghajtóra másoljuk ki a stringeket (egy ide, egy oda), majd összefésülésnél a másik kettőre az immár dupla hosszúakat, és előbb-utóbb egybeolvad az egész. Ám mi van, ha egy mágnesszalag-meghajtó épp nem működik? Hogyan oldható meg a sok adat rendezése három mágnesszalaggal?

A választ a Fibonacci-számok adják meg. Ha úgy pakoljuk ki a stringeket, hogy a két töltött mágnesszalagra mindig annyit pakoljunk pluszba, amennyi a másikon már van, akkor a Fibonacci-számsor – 1, 1, 2, 3, 5, 8, 13, 21, 34 stb. – két szomszédos eleménél fejezzük be a

kipakolást, és ha erről elkezdjük az üresre összemerge-olni a stringeket, akkor, amikor a kisebbik kiürül, a másik kettőn a Fibonacci-sorozat alacsonyabb sorszámú tagjai lesznek. Így a merge folytatódhat egészen 1-1-ig, és ezt a kettőt összemerge-ölve rendezve lesz a teljes adattömeg.

Nekem ez a takarékos összefésülés annyira megtetszett, hogy már a belső rendezésnél is alkalmaztam a Fibonacci-számok alapján történő sorba rendezést, egyre nagyobb rendezett halmazokat kialakítva, egészen addig, amíg a memóriakorlát miatt a rendezett sort – stringet – ki nem kellett írni az egyik mágnesszalagra.

A feladat mellett azt is szerettem, hogy a Videoton 1010B mellé leülhettünk, akár bele is kulcsolhattunk bináris kódsorokat, megszűnt az a távolság a gép és a programozó között, mint ami mondjuk még a SZÜV GIER gépénél megvolt. Amit nem szerettem, az a hadiüzem jelleg: a gyárkapuban álló, a kerítés mellett grasszáló géppisztolyos őrök, ezek miatt munkába érkezéskor, munkából való távozáskor mindig gyomorszorításom volt. Ezért aztán – amikor a Videoton eladott egy VT 1010B számítógépet a fehérvéri Volánnak, felkerestem Varga Sándor gazdasági igazgatóhelyettest és Escher József adatfeldolgozási főosztályvezetőt, hogy nem kell-e nekik egy olyan szakember, aki ért a programozásához. Kellett. Ezzel – fél év után – véget ért a Videoton-os korszakom.

A Volánnál rám bízta a programozók vezetését, Elek Gyula villamosmérnök barátomra pedig a gépterem vezetését, hozzátartoztak természetesen az operátorok is (bár nekik volt egy köztes főnökük Salla Éva személyében). Gyulán és rajtam kívül a fiatal lelkes kollegák nem sokat értettek a számítástechnikához, így az első hónapok képzéssel teltek el. Mivel várható volt, hogy folyamatosan bővül majd a létszám, úgy gondoltam, érdemes két tankönyvben is rögzíteni az átadandó tudást, ezért megírtam a *Videoton 1010B programozási kézikönyvet* (1. függelék), majd – a géptermi dolgozók támogatásához – a *Videoton 1010B operátori kézikönyvet* is. Varga Sándor gazdasági igazgató-helyettes el is dicsekedett ezekkel Kázmér Jánosnak, aki akkor egyszerre vezette a Videoton Számítástechnikai Gyárat Székesfehérváron és a Videoton Fejlesztési Intézetet Budapesten. Kiderült, hogy eddig ilyesmit a Videoton nem tudott szállítani a gépek mellé, és a két vezető meg is egyezett abban, hogy az elkészült műveket a továbbiakban hivatalos kézikönyvként mellékelhetik a gépek mellé. Akkoriban mindez a szerzőt nem nagyon érintette, hisz az általam munkaidőben írt könyvek szerzői joga a megírás pillanatában átszállt a Volán 14-es számú Vállalatra, így nekem ettől csak a keblem dagadhatott, a pénztárcám nem. (A könyveknek sem az élén, hanem az utolsó oldalának legalján jelent meg a nevem: „Az összeállításért felelős: Vértés János”.)

Hogy mit mutatnak ma számunkra ezek a könyvek? Például azt, hogy miképp programoztunk anno.

Nos, talán nagyobb körültekintéssel és előkészítéssel, mint mondjuk a PC-s korszakban. A *Programozói kézikönyv* első 29 oldala a feladatelemzéssel, azon belül a funkcionális és organikus analízissel foglalkozik. Amikor a rendszerszervező/programozó megtervezi a feladatok elemi műveletek láncolatára való bontását, akkor olyan mélységben gondolja át az algoritmust, mint majd maga a programsorokat értelmező és feldolgozó gép. Manapság, amikor a mesterséges intelligenciának egy összetett mondatban is kiadhatjuk a feladatot – például azt, hogy rendezd nagyság szerint növekvő sorba a beadott három képen található számokat –, bele sem gondolunk (bele sem kell gondolnunk), hogy ehhez miképp követik egymás az összehasonlítások, döntések, programlépések. A hetvenes évek elején még – és ez nemcsak a gépi kódú programozásra igaz, hanem a magasabb szintű programnyelvek használatára is – a feladatelemzés során a rendszerszervezőnek, a program-analitikusnak és a programozónak gyakorlatilag mindent kellett tudnia, amit majd a programjuk is elvégez. A számítógép előnye „mindössze” az volt, hogy a megtervezett (és általában sokszor ismétlődő) műveleteket a gép sokkal gyorsabban el tudta végezni (már akkor is), mint az ember.

Az is jelentős eltérés, hogy akkoriban egy programozónak azt is meg kellett tanulni, hogy a perifériákon milyen struktúrában helyezkednek el az adatok, hogy mi jelzi, ha, mondjuk, egy mágnesszalagon több fájl van. Ráadásul még arra is kellett egy kis szubrutint, programcskát írni, hogy ellenőrizzük: a megfelelő mágnesszalagot tette-e fel az egységre az operátor. Hogy miképp nézett ki ez az ellenőrzés? Így (most folyamatábra helyett elégedjünk meg a folyamatleírással):

1. A program kiírja a konzolírógépen a HDR1 üzenetet, és várja, hogy az operátor beírja a fájl 15 karakteres nevét.
2. A program beolvassa a mágnesszalagról a HDR1 blokkot, és annak 5-19 karakterét (az első 4 HDR1 ott is) összehasonlítja a beírt karakterekkel.
3. Ha stimmel, a program az első adatblokk elé pozicionálja a mágnesszalagot, ha eltérés van, akkor HDR1 után kiírja azt a nevet, amit a mágnesszalagon talált, visszatekericseli a mágnesszalagot, megvárja a mágnesszalag cserét és újra visszaugrik az első lépéshez.

A „fél évszázada” és a „ma” közötti programozás legnagyobb különbsége azonban a takarékoság: spórolni kellett a memóriával, spórolni kellett a gépidővel. Ha mesterséges intelligencia még nem is volt, de problémaorientált programnyelv már igen, erre – az ALGOL-hoz közelálló – nyelvre volt a legkönnyebb átültetni a feladatanalízis során megfogalmazott

algoritmust, ugyanakkor az így megírt program futtatási előkészítése, végrehajtási ideje és memóriaszükséglete (általában jóval) nagyobb volt, mint a gépkódban írotté. És bár a gépkódban írott program kevésbé volt áttekinthető, nehéz volt javítani, átírni, de a Videoton 1010B korlátozott erőforrásai sok esetben kényszerítették ki, hogy ezen az úgynevezett alacsony szinten készüljenek a programok. Ráadásul az, aki az utasítások 8 bites alakját is ismerte, könnyedén javította a lyukszalagot (többnyire ez volt a programok adathordozója), vagy akár a kezelő panelen található kulcsokkal is be tudott vinni a memóriába egy néhány utasításból álló programocskát.

A Volánnál az első programokat a gépterem felavatásához kellett igazítani:

- Megrendelés No. 1: „Fiúk, majd itt – a gépterem ajtajában – nem nemzeti színű szalagot kéne elvágni, hanem lyukszalagot. De nem akármilyen lyukszalagot, hanem egy olyat, amelyen a lyukak azt a feliratot adják ki, hogy Videoton – Volán – Videoton – Volán – Videoton – Volán... Meg tudjátok csinálni?”
Hát persze. Megcsináljuk.



A Székesfehérvári Volán géptermeének felavatása, 1973.¹

¹ A fehér köpenyesek balról jobbra: Elek Gyula gépteremvezető, Salla Éva operátor-csoportvezető és Vértes János programozó-csoportvezető. A szalagot Tapolczai Kálmán, a Volán Tröszt vezérigazgatója vágja át. Bal válla mögött szerényen a háttérbe húzódva Kázmér János, aki akkor a Videoton Számítástechnikai Gyár és a VIFI igazgatója volt, később lett az egész Videoton Rt. elnök-vezérigazgatója. A kép jobb szélén Demeter Béla, a Videoton műszaki igazgatója.

- Megrendelés No. 2: „A lyukszalag elvágása után a miniszter elvtárs odamegy a géphez, rámutatunk, hogy nyomja meg a MARCHE-gombot, amire a gép elkezd kiírni a displayre, hogy szeretettel köszönti a megjelenteket a Videoton 1010B... és akár be is mutatja magát. Menni fog?”

Ehhez már kellett egy aprócska trükk is, a programot mi előre elindítottuk, de volt benne egy leállító HLT utasítás, amin a MARCHE-gomb újbóli megnyomásával lépett túl, és végezte el a kiírást.

- Megrendelés No. 3: „Az lenne jó, ha a gép hangosan fel is olvasná azt a szöveget, amit kiír a kiírással szinkronban.”

Na, erre már próbáltuk megmagyarázni főnökeinknek, hogy ez a gép még nem tud beszélni, de ők nem nagyon örültek az ellentmondásnak, mi meg nem szerettük kimondani, hogy lehetetlen. Ezért aztán Salla Éva szépen lassan, a kiírás ütemében magnóra mondta az üdvözlő és bemutatkozó szöveget, és megegyeztünk, hogy amikor a miniszter megnyomja a MARCHE-ot, akkor az üvegfal mögött Salla Éva elindítja a kihangosított magnót. Hát így. Kevesen tudják, hogy a Videoton 1010B már beszélni is tudott 1973-ban. Legalábbis a fehérvári Volánnál.

Az új gépet egy meglévő informatikai rendszerbe kellett integrálni: a Volán vállalatoknál már évekkel korábban lyukkártyákra rögzítették (Aritma típusú berendezésekkel) a menetleveleket, ezeket buszsofőrökkel küldték el Budapestre a Volán Elektronikához (ez még a cég hőskorszaka volt dr. Tápay Tamás vezetésével), ahol Univac gépeken dolgozták fel a lyukkártyákból kinyert adatokat. A vállalatok ugyancsak „on busz” érkező válaszüzenetben, leporellón kapták meg a feldolgozott – bérszámfejtéshez, könyveléshez és egyéb belső folyamatokhoz szükséges – információt.

Induláskor a Volán Elektronikánál úgy gondolták, hogy ezt a folyamatot teheti picit korszerűbbé a Volán 14-es számú Vállalathoz telepített Videoton 1010B. A kártyákat (vagy akár lyukszalagokat, hiszen az új technológia mellett már arra is lehetett rögzíteni az adatokat) itt helyben mágnesszalagokra lehetett vinni, a mágnesszalagok vándoroltak a menetrendszerű buszokon fel a fővárosba és vissza, és itt helyben lehetett kinyomtatni a feldolgozások eredményét.

A fehérvári Volán vezetői azonban ambiciózusabbak voltak annál, mintsem, hogy az ide telepített gép csak a Volán Elektronika Univac gépparkjához nyújtson inputot és outputot, jöttek önálló feladatok is. Volt Fehérváron egy komoly diagnosztikai állomás, ahol a teherautók fődarabjairól készült értékelés. Ezen adatok összevetése a menetlevelekből kinyert kilométer-

adatokkal jó alapot nyújtott ahhoz, hogy prognosztizálni lehessen, mikor kell, mondjuk, az YN-1528 forgalmi rendszámú ZIL-en kicserélni valamelyik fődarabot. Ha akkoriban lett volna EU-s pályázat, erre biztos kiharcoltak volna a vállalat vezetői néhány tízmilliót, de azt hiszem enélkül is ment a pénzek kiudvarlása a minisztériumból: amikor a miniszterhelyettes elvtárs ellátogatott hozzánk az igazgató elvtárs meghívására, még a diagnosztikai állomást is újramázolták olajfestékkel, mert bár a téma a szoftver volt, azért a felvezetés mégiscsak a műhelyben indult.

Kedvenc projektem az „utazószemélyzet gépi vezénylése” volt. Ez lényegében a buszsofőrök és kalauzok (több mint 300 ember) havi munkabeosztását takarta, ki mikor melyik járaton dolgozzon. Amellett, hogy egy többszáz soros és többszáz oszlopos mátrixot kellett optimalizálni, számos peremfeltételnek is meg kellett felelni: a csuklós buszt csak csuklós vizsgával rendelkező sofőr vezesse; két műszak között legyen meg a törvény által előírt pihenőidő; az abdai dolgozó lehetőleg úgy kezdjen, hogy az Abda-Dunaújváros járatot hozza be Dunaújvárosba, és úgy végezzen este, hogy egy busznyi utassal kimegy Abdára; mindenkinél kijöjjön a havi munkaidő úgy, hogy mínusz ne legyen, de sok túlórát se kelljen fizetni stb., stb.

Ebben a feladatban legalább volt némi matematika, Simplex-módszerrel lehet ezt a mátrixot optimalizálni. Csakhogy nagyon nagy mátrixról és nagyon kis teljesítményű gépről beszélünk (na jó: az ENIAC-hoz képest gyors volt, de mondjuk egy mai telefonnal már egy napon említeni sem merném). Ezért aztán a korábban már említett takarékosagra való törekvés itt már a tervezés szintjén belépett. Tudtuk, hogy nem a teljes mátrixot kell optimalizálni, hanem annak csak egy részét, hiszen a kötöttségek egy része meg is szabta, hogy egy adott járaton ki lehet, vagyis ezeken nem volt mit optimalizálni, Végül egy szűkített mátrixon a Vogel-Kordamódszer meghozta azt az eredményt, amit a teljes mátrixon a Simplex hozott volna, csak épp az előbbihez az utóbbinál jóval kevesebb erőforrás kellett, elég volt a 1010B mintegy két gépórája (ugyanazt beprogramozta a Volán Elektronika is, náluk a Univac – Simplex-módszerrel – 30 órán át zümmögött).

Büszkén vittem ki az első gépi vezénylési leporello-táblákat Orgovány Pista bácsihoz, a buszpályaudvar főnökéhez, aki hosszasan nézegette, majd megszólalt:

- Nem jó.
- Mi nem jó, Pista bátyám?
- Az egész. Például a Józsi szombaton disznóvágásra megy, ti meg beosztottátok.

- De Pista bátyám, előzetesen megpróbáltuk felmérni az összes korlátozó körülményt, simán belefért volna, hogy beállítsuk ezt a korlátot is, és akkor a program nem osztotta volna be szombatra Józsit. Lefuttassuk újra?
- Futtassátok! De a legjobb lenne, ha a számítógép egy olyan táblát csinálna, amin ott vannak sorban a nevek a bal szélén, és mellettük harmincegyszer az, hogy de/du, de/du, de/du... Majd én bekarikázom, hogy az illető mikor jön be délelőtt, mikor délután, és beírom a járatszámot.
- De Pista bátyám, ehhez nem kéne számítógép, egy ilyen papírt egy nyomdagép is ki tudna nyomtani előre, egész évre... Épp az a lényeg, hogy a számítógép optimalizáljon...
- Nekem ne optimalizáljon, az az én dolgom. A nyomtatott papír meg azért nem jó, mert ha a Józsi mégsem akarom elengedni disznóvágásra, akkor, ha a számítógép csinálta a de-du-t, mondhatom neki, hogy a számítógéppel vitatkozzon, de egy nyomtatott papír esetén engem küld el melegebb égtájakra.

Később persze sikerült a kezdeti ellenállást letörni, én pedig írtam egy szép dolgozatot *Többet ésszel, mint programozással* címmel, amivel 1975-ben megnyertem azt versenyt, amit a KISZ által akkor indított „számítástechnikai védnökségi program” keretében írtak ki, s megkaptam a legnagyobb KISZ-kitüntetést, amit mások csak 20-25 évi mozgalmi munkával érdemeltek ki, és barátság-vonattal részt vehettem egy Kijev-Moszkva-Leningrád-Riga körutazáson, újra meglátogatva diákéveim néhány színterét.

Mindeközben cikkeket írogattam a *Fejér megyei Hírlap*ba, amelynek főszerkesztője egyszer egy hosszú beszélgetés keretében meggyőzött, hogy kettős énemnek a humán oldalát is ki kéne bontakoztatni, és néhány év programozás után ideje most már színház- és filmkritikákat írnom, vagy ha annyira kötődöm a számítástechnikához, akkor akár arra is módot kínál a *Hírlap*, hogy tudományos ismeretterjesztő cikksorozatban népszerűsítsem azt. Így lettem 1977-ben kulturális rovatvezető, picikét még jobban eltávolodva a matematikától. (Bár én a matematikát úgy definiálom, hogy nem a számolás és a számítás, hanem az absztrakció tudománya, és a lényeg elvonatkoztatására, a lényegtelen megkülönböztetésére az újságírásban is szükség van.)

Néhány év humán újságírás után azonban reál énem is ébredezni kezdett, így 1981-ben a korábbi munkahelyeimen született barátságok egy vállalkozást szültek: volt Videoton-os és Volán-os kollegáimmal megalakítottuk a KRONOSZ Számítástechnikai Társaság gmk.-t.

Azzal kezdtük, hogy Elek Gyula barátommal autóba ültünk, kimentünk Münchenbe, és beszereztünk az ott már kapható ZX81-ből egy példányt. Az 1 kilobyte memóriával rendelkező gép mellé vettünk egy 16K-s memóriabővítést, egy hőpapíros nyomtatót, (és – hogy saját épülésünk is meglegyen – egy sakkprogramot). 1100 márka volt a befektetésünk. Amikor mindezt összedugtuk lakásunkban egy TV-vel, és elkezdhattuk begépelni az első utasításokat, Videoton-os volt kollegám összecsapta a kezét: „Te most tíz perc alatt installáltál egy számítógépet!” A Videoton 1010B-vel – a duplapadlós, álmennyezetes gépterem kialakítását nem is számolva – ez három hét!

Igen, ez volt számunkra a személyi számítógépek korszakának kezdete, még ha akkoriban ezek a valóban otthoni játékszerek nem a háztartásokban, hanem olykor nagyvállalatoknál jelentek meg. A ZX81-hez (kihasználva, hogy ehhez mágnesszalagos adattároló is csatlakoztatható volt egy kazettás magnó képében) például BASIC-ben készítettünk egy nyilvántartó programot *Katalógus* néven. Ezt néhány példányban el is adtuk cégeknek, de sokkal nagyobb sikert ért el ugyanennek a Commodore 64-re írt verziója, amire nem sokkal a ZX81-es korszak után mi is áttértünk, és a cégek is elkezdtek arra építeni informatikai hátszágukat. A *Katalógust* kiegészítettük egy szövegszerkesztővel is (ezt *Korrektornak* hívtuk), és egy egyszerű táblázatkezelővel (ez volt a *Kalkulátor*). Ezeket a programokat már floppyn árultuk egy kis füzet hátsó borítójára ragasztva, a füzetben a használati utasítással. Akkoriban a HVG-ben egy egyhatod oldalas hirdetés 19.800 Ft-ba került. Azt mondtuk: egy program legyen 9.800 Ft, a 3K együtt 19.800 Ft. És addig hirdetjük a HVG-ben, amíg ennek hatására a következő héten egynél több megrendelés érkezik.

A 3K-t így közel hat hónapig hirdettük és árultuk, addigra már gyakorlatilag minden Commodore 64-re rákerült (vagy az eredeti vagy egy másolat), így nem lehetett többet eladni. A mérleg: 2,5 millió bevétel, cca. fél millió hirdetési költség. A nyolcvanas évek első felében ezt 3 kis BASIC-programért egyértelműen sikernek lehetett elkönyvelni.

Közben persze megjelentek a PC-k és a PC-klónok, így a Kronosz Számítástechnikai Társaság fénykorában már ezekre fejlesztettünk egész komoly ügyviteli rendszereket, például a győri Ingatlankezelő Vállalatnak raktárnyilvántartást, a MOL elődjének (az ÁFOR-nak) bérszámfejtést.

Ez már semmiképp nem a hőskor, így a régmúlt felidézését be is fejezhetném, de ha már személyes emlékszilánkokról szól ez az írás, még lekerekítem a folytatással. 1986-tól – amikor elindult a Computerworld Informatika Kft. (a későbbi IDG elődje), úgy éreztem, hogy ez az a hely, ahol újságíró- és informatikai énem egyszerre érvényesülhet, úgyhogy informatikai

újságíró lettem. A CWI-ből kiválva pedig vállalkozó, 1991-től saját kiadót alapítottunk, ám a kétezres évek elején a *VGA Monitor*, az *Internet kalauz*, a *Business Online* újságjainkat lassan, de biztosan megölte az internet. Szerencsére addigra az újságok online hátteréhez néhány szoftvert is kifejlesztettünk, egy CMS-rendszert, egy hirdetéskiszolgáló rendszert, így ezek értékesítésére helyeztük át a fókuszot, és létrehoztuk az Adverticum Zrt.-t, amely újra egy fejlesztő és online szolgáltató cég. És ma megint fel van adva a lecke: hogyan építsük be a mesterséges intelligenciát a rendszerünkbe.

Vagy kérdezzük meg erről magát az AI-t? Talán még a kódolást is elvégzi helyettünk. Mert ma már így megy a programozás. Kicsit egyszerűbben, mint régen.

2025. szeptember

Függelék:



1. Feladatelemzés, program analízis -----	1.
1. Algoritmus -----	3.
1.1. Belső műveletek -----	5.
1.1.1. A program tömörítése (térbeli optimalizálás) -----	5.
1.1.2. A szubrutin fogalma -----	6.
1.1.3. Paraméter átadás -----	7.
1.1.4. Gépidő csökkentés -----	7.
1.2. Perifériális műveletek -----	9.
1.2.1. A szimultaneitás -----	9.
1.2.2. File kezelés -----	10.
2. Perifériák adminisztrációja -----	14.
2.1. Mágnesszalag -----	14.
2.2. Minidisk -----	18.
2.3. Kártyaolvasó -----	18.
2.4. Karakteres perifériák -----	18.
3. Elemzési példa: Adatrendezés -----	20.
3.1. EXTERN algoritmusok -----	20.
3.2. Az EXTERN belső műveletei -----	22.
3.3. Az INTERN belső műveletei -----	25.
3.3.1. Fibonacci rendezés -----	25.
3.3.2. Helycserés rendezés -----	25.
3.3.3. Bővíthető szelektív rendezés -----	26.
3.4. Az INTERN perifériális műveletei -----	27.
4. Összefoglalás -----	29.
II. Programnyelvek -----	30.
1. Gépi szintű értelmezés -----	31.
1.1. Bevezetés a számítógép címzési rendszeréhez -----	31.
1.2. Az utasítás végrehajtása -----	32.
1.3. Assembler nyelvek -----	33.
1.3.1. ASTROL (szintaktika) -----	33.
1.3.2. VIDAS (szintaktika) -----	38.
1.4. A VIDEOTON 1010B utasításkészlete -----	44.
1.5. Perifériák programozása gépi szinten -----	47.
1.5.1. A VT 1010B perifériális és megszakítási rendszere -----	47.
1.5.2. Programozott csatorna (LP) -----	48.
1.5.3. Standard csatorna (LS) -----	55.
1.5.4. Direkt csatorna (LD) -----	63.
2. Értelmező szintű programozás -----	66.
2.1. ASTROL standard mikroprogramkönyvtár- és az interpretatív programozás -----	66.
2.1.1. Szubrutinhívás -----	66.
2.1.2. Szubrutinhívás szubrutinból -----	67.
2.1.3. Standard mikroprogramkönyvtár -----	68.

I. Feladatelemzés, program analízis -----	1.
1. Algoritmus -----	3.
1.1. Belső műveletek -----	5.
1.1.1. A program tömörítése (térbeli optimalizálás) -----	5.
1.1.2. A szubrutin fogalma -----	6.
1.1.3. Paraméter átadás -----	7.
1.1.4. Gépidő csökkentés -----	7.
1.2. Perifériális műveletek -----	9.
1.2.1. A szimultaneitás -----	9.
1.2.2. File kezelés -----	10.
2. Perifériák adminisztrációja -----	14.
2.1. Mágnesszalag -----	14.
2.2. Minidisk -----	18.
2.3. Kártyaolvasó -----	18.
2.4. Karakteres perifériák -----	18.
3. Elemzési példa: Adatrendezés -----	20.
3.1. EXTERN algoritmusok -----	20.
3.2. Az EXTERN belső műveletei -----	22.
3.3. Az INTERN belső műveletei -----	25.
3.3.1. Fibonacci rendezés -----	25.
3.3.2. Helycserés rendezés -----	25.
3.3.3. Bővíthető szelektív rendezés -----	26.
3.4. Az INTERN perifériális műveletei -----	27.
4. Összefoglalás -----	29.
II. Programnyelvek -----	30.
1. Gépi szintű értelmezés -----	31.
1.1. Bevezetés a számítógép címzési rendszeréhez -----	31.
1.2. Az utasítás végrehajtása -----	32.
1.3. Assembler nyelvek -----	33.
1.3.1. ASTROL (szintaktika) -----	33.
1.3.2. VIDAS (szintaktika) -----	38.
1.4. A VIDEOTON 1010B utasításkészlete -----	44.
1.5. Perifériák programozása gépi szinten -----	47.
1.5.1. A VT 1010B perifériális és megszakítási rendszere -----	47.
1.5.2. Programozott csatorna (LP) -----	48.
1.5.3. Standard csatorna (LS) -----	55.
1.5.4. Direkt csatorna (LD) -----	63.
2. Értelmező szintű programozás -----	66.
2.1. ASTROL standard mikroprogramkönyvtár- és az interpretatív programozás -----	66.
2.1.1. Szubrutinhívás -----	66.
2.1.2. Szubrutinhívás szubrutinból -----	67.
2.1.3. Standard mikroprogramkönyvtár -----	68.

	Oldalsz
2.2. A monitor mikroprogramkönyvtára és rezidens alprogramjai -----	77.
2.2.1. ASTROL program monitor módban -----	77.
2.2.2. A monitor rezidens mikroprogramjai -----	80.
2.2.3. A monitor rezidens alprogramjai -----	87.
2.3. VIDOS-Extrakódok és makrók könyvtára, FIOCS rendszerrutinok -----	90.
2.3.1. Extrakódok -----	90.
2.3.1. Makróutasítások -----	96.
2.3.2. Rendszerrutinok -----	97.
3. VIDOS operációs rendszer -----	99.
3.1. Szubrutin kezelő rendszer (SKR) -----	99.
3.1.1. Szubrutin fej -----	100.
3.1.2. Paraméterkezelés -----	101.
3.1.3. S direktívák -----	102.
3.1.4. T direktívák -----	105.
3.1.5. A szubrutin törzse -----	106.
3.1.6. Kilépés a szubrutinból -----	106.
3.2. Könyvtárépítés, rendszerkarbantartás -----	108.
3.2.1. Könyvtárak -----	108.
3.2.2. Katalógustételek -----	109.
3.2.3. Könyvtárépítés -----	110.
3.3. Fizikai input-output rendszer -----	111.
3.3.1. CCB tábla -----	111.
4. Példa programok -----	115.
4.1. Mágnesszalagos file nevének ellenőrzése - ASTROL és Monitor inter- pretativ nyelven irt programban -----	115.
4.2. ASCII kódú decimális aritmetika beépítése a VIDOS Extrakód könyvtá- rába -----	12.

Mellékletek:

1. sz Memoria "0" lap -----	12
2/a. sz ASCII kód -----	13
2/b. sz ASCII kód display-re -----	13
3. sz EBCDIC-kód -----	13
4. sz Fordítási hibajegyzék -----	13
5. sz Számrendszerek -----	13