

TANÍTSUNK-E PROGRAMOZÁST NEM INFORMATIKA SZAKOS HALLGATÓKNAK IS?

Timár Lajos, TimarL@almos.vein.hu
Czihlár Helga, cc04hc01@triton.pluto.vein.hu
Veszprémi Egyetem, Matematikai és Számítástechnikai Tanszék

Abstract

To teach fundamentals of programming or not to teach for non science students that is the question underlying the meditation you can read in our paper. Experiences gained during our courses on application programmes have shown that some programming basics may be useful even for word processing. As for data base handling, you can not build any user-friendly application without knowing the fundamentals of programming. We think that the lack of this knowledge is a good business for spreading icon- and mouse-click-based software products and expensive three-day courses.

1. Bevezetés

Vegyéssz mérnök PhD hallgatók közül többen is "rájöttek" arra, hogy azoknak a programozási ismereteknek, amit egyetemünkön szereztek semmi hasznát (?) sem veszik leendő munkájában. Vagyis: programozást nem kellene (?) tanítani mérnökjelölteknek sem. Számítástechnika tanárszakos hallgatók egy része zúgolódik azért, mert túl soknak tartja a programozási képzést. Inkább foglalkoznánk helyette mással, például káoszelmélettel, multimédiával.

Mindig elgondolkoztató az alu Iról jövő kritika. A számítástudomány rohamos fejlődése nagy kihívás, amivel lépést kell tartani. Ugyanakkor nagy a kísértés is. Vajon nem fölösleges időtöltés a programozás akkor, amikor ez idő alatt az információs szupersztrádán néhány egérekattintással bejárhatjuk az egész világot és előbb-utóbb értékes információhoz is juthatunk. Másfelől: ma már minden szakterületen létrehozták azokat a szoftvereket, amelyeket eddig a szakterület művelőinek kellett megírnia.

A kritikákban és a kísértésekben rejlő félti gagságok mellé hadd tegyük saját féligagságainkat:

- programozási alapismeretekkel nem rendelkező hallgatóknak a szövegszerkesztést nehezebb megtanítani, mint programozási alapismeretekkel rendelkezőknek;
- programozási alapismeretekkel nem rendelkező hallgatóknak adatbáziskezelést jól megtanítani csaknem reménytelen vállalkozás.

Az előadásban - saját és mások tapasztalatai alapján - megkíséreljük bebizonyítani a programozási alapismeretek fontosságát és nélkülözhetetlenségét. Meggyőződésünk, hogy programozási alapismeretek nélkül még szövegszerkesztést sem lehet hatékonyan megtanulni. Így a kérdés számunkra nem az, kell-e programozást tanítani mérnök vagy nem mérnök hallgatóknak. A kérdés az, hogy milyen mértékben és hogyan. Reméljük, hogy az előadásban felvetődő gondolatok heves vitát váltanak ki, amit hasznosíthatunk oktatásunk színvonalának emelésében.

2. Az egérekattintások száma napról-napra nő...

"Ma csupán ezekből az egérekattintással működő népszerű web-oldalakkból van már több mint százmillió, és számuk napról-napra nő." Az idézet egy dunántúli számítógépes cég vezetőjével készített beszélgetésből való.

Korunk nagy kísértése: néhány egérekattintással szerkeszthetsz szöveget, "letölthetsz" adatbázisokat, vagyis alkalmazói programok használatát bárki könnyen és gyorsan elsajátíthatja, programozási alapismeretek nélkül is.

Tényleg elég néhány egérekattintás? Ilyen igénytelen lenne egy felhasználó? Nem kíváncsi arra, milyen parancsok és milyen sorrendben hajtódnak végre? Sajnos a tisztességtelen irodalom erre nevel. Előfordul, hogy a "hogyan adjunk címet dokumentumunknak, ha a dokumentum megírása után jut eszünkbe?" - szövegszerkesztési kérdésre olyan elrettentő válasz jelenik meg meglehetősen drága könyvekben ¹, mint például:

- tartsuk lenyomva a CTRL billentyűt, miközben lenyomjuk a home billentyűt
- nyomjuk le az Enter billentyűt
- nyomjuk le a felfelé-nyíl billentyűt
- a festékszalag font-menüjéből válasszuk ki a Helvetica betűtípust
- a font-méret legördülő menüből válasszuk ki a 14 pontos méretet
- írjuk le a címet
- a shift billentyűt lenyomva tartva nyomjuk le az ENTER billentyűt
- írjuk le a szerző nevét
- kattintsunk az eszközménüsor középreigazító ikonjára

Csoda-e, ha ilyen válaszok hatására a számítástechnikai alkalmazásban annyira fontos absztrakciós képesség csírája is elpusztul a felhasználóban. Kénytelen bemagolni a hosszú parancssorozatot, anélkül, hogy tisztában lenne: mit miért tesz. Mi történik, ha elfelejti a 6. műveletet? Mihez kezd a felhasználó, ha a billentyűzetén Return felirat van az Enter helyett. Még rágondolni is rossz! Mennyivel könnyebben memorizálható egy lényegre törő - az apró részleteket elhallgató - lépéssorozat:

- *szúrjunk be* néhány üres sort a dokumentum első sora elé
- az üres sorok egyikébe írjuk be a címet és új sorba, de *ugyanabban a bekezdésben* maradva, írjuk be a szerző nevét
- valamilyen eszközzel igazítsuk középre a címet és a szerzőt
- az igényeknek megfelelően változtassuk meg a betűtípust (pl. Times Roman)

Vagyis: inkább azt jegyezzük meg, mit kell tenni, mint azt, hogy hogyan.

A beszúrás, bekezdés és karakterformás alapműveletek felesleges leírásával nem terejük el a figyelmet a lényegről. Ezzel a felhasználót sem korlátozzuk abban, hogy egy formázási műveletet milyen eszközzel hajtson végre (menüvel, eszközsorral, billentyűkombinációval).

Természetesen ez az egérekattintásos, látszólag felhasználóbarát szemlélet egyben kihívás is. Időről időre kénytelenek vagyunk átalakítani megszokott tematikánkat úgy, hogy az egérekattintást előkészítő alapozási idő rövidebb legyen. Mindezt azért, hogy a hallgató minél előbb szakemberként és kellő kritikai érzéssel száguldhasson az információs szupersztrádán. Más szóval: saját maga is tudjon web-oldalakat készítenie és csak akkor készítsen ilyeneket, ha valóban szükség van rá és ha már szükségét látja, akkor jó web-oldalakat készítsen. ²

¹Egy budapesti könyvesboltban tanulmányoztam egy hasonló, kérdés-felelet könyvet (címére nem emlékszem). Elrettentő példának majdnem megvásároltam. Sajnos az ára is elrettentő volt.

²Lemay L.: Teach Yourself Web Publishing with HTML in a Week. SAMS Publ. 1995.

3. Egy vagy ezeregy éjszaka?

Kell-e programozási alapismeret szövegszerkesztéshez? Sőt, kell-e egyáltalán tanítani?

“Nem kell, hiszen a lányom egyetlen éjszaka alatt megtanulta a WORD használatát”- hangzott el egy “Informatika az oktatásban” konferencia plenáris előadásában. Vannak hallgatók is, kollegák is, akik lealacsonyítónak tartják, ha egy adjunktus vagy docens szövegszerkesztést tanít. Az elmúlt évben egy barátomat tanítottam szövegszerkesztésre. A huszadik óra után kicsit szomorúan jegyezte meg, hogy ő egy reménytelen eset. Kollégája egyetlen éjszaka elsajátította mindazt, ami neki hónapos munkával sem sikerült. Kértem, hogy mutasson néhány olyan dokumentumot, amit kollégája írt. Természetesen nem kinyomtatva, hanem DOC-file-ként. Gyanúm igazolódott. A dokumentum sok-sok egysoros bekezdést, felesleges szóközöket és tabulálásokat tartalmazott. Barátom kicsit megnyugodott és büszkén mutogatta szakszerűen szerkesztett, új dokumentumait.

4. Főlölesleges időtöltés-e a programozás?

Többéves oktatási gyakorlat alapján határozottan állíthatom, hogy azok a hallgatók, akik programozási alapokkal is rendelkeznek, még a szövegszerkesztést is könnyebben, gyorsabban és főleg hatékonyabban sajátítják el, mint azok, akik sohasem programoztak. Azok akik sohasem programoztak nem igazán látják a különbséget a kétféle szemlélet között. Ők hajlamosak arra, hogy a technikai részleteket összekeverjék a feladatmegoldás alpműveleteinek ismétlődő, esetleg elágazó lépéseivel. A megszerzett szövegszerkesztési ismeret nem lesz tartós és nehezen tudják később hasznosítani a szoftver újabb verziójánál. Arról nem is beszélve, hogy egy másik cég (pl. Word Perfect) hasonló felépítésű és célú szoftverjével alig tudnak valamit kezdeni. Tételezzük fel, hogy az oktatási módszer jó, vagyis szövegszerkesztést tanulnak egy adott szövegszerkesztővel és nem egy adott szövegszerkesztőt (összhangban egy Békéscsabán rendezett “Informatika az oktatásban” konferencia vitafórumain elhangzottakkal).

A lényegkiemelő képességeket legjobban a programozás fejleszti. Erre példa egy angol szakos csoport, amelyben néhányan fél évig számítástechnika szakosok is voltak. A számítástechnika szakosok fél évig programozással foglalkoztak (algoritmizálás). Magam sem gondoltam, hogy ez az előélet a szövegszerkesztésben nagy előnyt jelent majd a többiekkel szemben. Nem annyira az előny, hanem annak mértéke lepett meg. Különösen a táblázatokban és a körlevelekben nyújtottak kiemelkedő teljesítményt.

A félreértések elkerülése végett hangsúlyozom: szövegszerkesztést programozási ismeretek nélkül is jól el lehet sajátítani. Erre az említett angol csoportban is volt példa. Csupán azt állítom, hogy programozási alapismeretek birtokában a szövegszerkesztés gyorsabban megtanítható és megtanulható. Továbbá az ilyen alapozással elsajátított szövegszerkesztési tudás más, hasonló rendeltetésű szoftverek megismerésekor később is hasznosítható. Ráadásul önképzéssel, költséges tanfolyam nélkül.

Mi az, ami egy rövid programozási gyakorlatból a szövegszerkesztésben hasznosítható? Nem a programnyelv (pl. PASCAL), hanem a feladatmegoldásokban kialakult vagy továbbfejlesztett lényegkiemelő készség és képesség, valamint olyan, nyelvtől független alapelemek mint például: adatok, aritmetikai és logikai műveletek, feltételes elágazások, ciklusok és alprogramok. Aki tisztában van ezekkel a fogalmakkal, annak egy adott szövegszerkesztési feladat elvégzéséhez csak lényeges lépéseket kell megjegyeznie. Például: hogyan adjunk címet egy táblázatnak utólag?

- tegyük a táblázat másolatát a Clipboard-ba
- szerkesszük meg a címet a kívánt betűfajttával és típusal és igazítsuk középre
- gépeljük egy vagy több üres sort a cím után
- hozzuk vissza a táblázatot a Clipboard-ból

5. Megtanulható-e az adatbázis-kezdés programozási ismeretek nélkül?

Mit tanítasz? Access-t? - kérdezte nemrég egy nem informatikus kolléga. Nem, adatbáziskezelési alapokat az Access felhasználásával. Természetesen az Access-t is megismerjük valamennyire - próbáltam

válaszolni a kérdésére. Nem tettem hozzá, hogy csak gyakorlati órák formájában ez szinte reménytelen vállalkozás nem számítástechnika szakosoknak. Órákig kellene beszélni a táblázatokról, az egy-egy és egy-sok ;s sok-sok kapcsolatokról. Az adatok típusáról és a kulcsokról. Kivételt képeznek a már említett számítástechnikai előélettel rendelkezők. Számukra néhány percben is sikerül megmagyarázni az adatmodellezési alapfogalmakat. Programozási és adatmodellezési ismeretekkel rendelkező szakos hallgatóknak viszont egy adat-báziskezelő program minden szépségét feltárhatjuk (Form, Report, Macro, saját nyelven írt alprogramok), éspedig aránylag rövid idő alatt.

Sok olyan egykori mérnök-hallgatómmal találkoztam, akik visszanézve számítástechnikai alapképzésükre szükségtelennek és fölöslegesnek érzik a programozással eltöltött fél évet. Helyette inkább táblázatkezelést vagy adatbáziskezelést kellett volna tanítani - mondogatják. Abban teljesen igazuk van, hogy adatbázis-kezelést tanítanunk kellett volna. Abban viszont nincs igazuk, hogy programozási alapismeretek helyett. Úgy érzik, hogy semmi hasznát sem veszik a programozási ismereteknek, mert nem írnak programot. Lehet, hogy a megtanult nyelvet is elfelejtették. Mindazonáltal észre sem veszik, hogy egy-egy adat-báziskezelő műveletsorozat macro-vezérelt együttműködésének kidolgozásában milyen előnyt jelentenek olyan alapfogalmak (tudat alatti?) ismerete mint például: feltételes elágazás, ismétlődő szerkezetek, alprogramok használata, adattípusok stb.

Tapasztalataim alátámasztják a programozási alapismeretek hasznosságát. Tanítottam adatbázis-kezelést (WORKS, ACCESS) programozási alapismeretekkel nem rendelkezőknek (egyszakos angol) és programozási alapismeretekkel rendelkezőknek (pl. számítástechnika, kémia) is.

Legutóbb egy csoporton belül fordult elő mindkét típus. A társaság zöme olyan angol szakos, akik sohasem programoztak és nem is fognak. Kisebb részük fél évig számítástechnika szakos volt, vagyis rendelkezik némi programozási alapismerettel. Ők sokkal gyorsabban teljesítették a követelményeket, mint a többieknek. Továbbá igényesebb munkát is végeztek (ellenőrzött adatbevitel, felhasználóbarát felületek kialakítása stb.). A többiek programozási alapok (elmélet és gyakorlat) hiányában, a legnagyobb igyekezetük és igyekezetünk mellett sem tudták követni az egykor számítástechnika szakos társaikat.

Az adatbázis-kezelő szoftverek sikeres megismerése programozási ismereteken kívül adatmodellezési alapokat is feltételez (rekord, táblázat, kulcs, kapcsolattípusok, join stb.). Ha mindezzel rendelkezünk (pl. informatikus hallgatók), akkor van értelme az alkalmazói program felhasználói kézikönyveit tanulmányozni. Ez a MS Access esetében kb. 1500 oldalt jelent! Adatmodellezési előtanulmányok birtokában a számítástechnika tanár szakos hallgatók 10-12 hét alatt nyomógombos menüvezérelt, makrokkal szinkronizált űrlapokat kezelő alkalmazásokat hoztak létre. Ugyanakkor egy nem számítástechnika szakos csoporttal 30 óra alatt a táblázatkezelés (létrehozás és kapcsolatok) és lekérdezési alapokat sem sikerült megnyugtató módon elsajátítani.

Meggyőződésünk, hogy programozási ismeretek nélkül adatbázis-kezelést jól megtanítani reménytelen vállalkozás. Ez a megállapítás természetesen csak akkor lehet igaz, ha programozást tanítunk és nem programnyelvet (mellékesen azt is), valamint adatbázis-kezelést tanítunk és nem egy adatbázis-kezelő alkalmazói programot (mellékesen azt is) [1,2].

6. Jó üzlet-e a programozási ismeretek hiánya?

A szoftverterüjdonságokról szóló hirdetések egy része azt sugallja, hogy a ma az informatika alkalmazójának egyáltalán nem kell programoznia, hiszen

- programozás ismeretek nélkül is készíthetünk alkalmazói programot (pl. programgenerátorok)
- néhány egérgattintással is készíthető program...(pl. HyperCard)

Másik részük viszont arról győzi meg az ügyfelet, hogy a programozás nem is olyan nagy dolog, hiszen

- Az a kevés kódot, amit egy felhasználónak meg kell írnia, elkészítik a varázslók (Wizzard).
- Intenzív tanfolyamon három nap alatt megismerhetjük a szükséges nyelvet (pl. WWW, script-ek)

Ilyen egyszerű lenne a kódolás? Egyetértek az Új alaplap 1995/5 cikkírójával, aki a programozás jövőjéről többek között így elmélkedik: "Nincs a mai értelemben vett programírás, csak ide-oda húzogatása ennek-annak a képernyőn az egerrel. A programozásnak kódolási részét valóban felváltja az új technika. De vajon az építőkockák, osztályok ellenőrzése, annak vizsgálata, hogy odaillesenek-e pontosan arra a helyre, ahová szánjuk őket, nem visz-e el több időt, s főleg nem követel-e a kódoláshoz szükségesnél akár mélyebb, alaposabb tudást?" [3]

Sajnos, a középiskolából érkező hallgatók jelentős részét "megfertőzik" az üzleti világ informatika hírei. Képtelenek megszűrni azokat és nehezen tudják elfogadni az alapozó tárgyak szükségességét. Azonnal Internet- és más multimedia-adatbázisokkal, objektum-orientált programozással és a felhasználói programok legújabb verzióival szeretnének foglalkozni.

E kihívásokkal (pl. WIN95 és Internet-függőség) akkor is kénytelenek vagyunk időről-időre lépést tartani, ha nem vagyunk meggyőződve az új irányzatok és termékek nélkülözhetetlen fontosságáról.

A programozási ismeretek hiánya jó üzlet, mert

- minden alkalmazói programverzió megjelenésekor új tanfolyamot lehet hirdetni
- ha a felhasználó rájön arra, hogy valami programozási ismeret jól jönne, akkor néhány napos, intenzív tanfolyamra jelentkezhet
- megveszik a legújabb verziót akkor is, ha az új szolgáltatásokat nem tudják kihasználni programozási ismeretek hiányában (pl. az új szolgáltatás új macrokat, és új függvényeket jelent), mert nem tudják eldönteni, hogy az új szolgáltatások programozási alapismereteket feltételeznek - programozási alapismeretek hiányában

Megjegyzem, a számítástechnika angol nyelvének nem ismerete még ennél is nagyobb üzlet, hiszen a "magyar embernek, magyar OFFICE-t!" jelszóval a tanulástól félők széles táborát lehet megnyerni. Ez a kérdés azonban megérne egy külön konferenciát.

7. Összefoglalás

- Az egérkattintás megkönnyíti a felhasználó munkáját, ha ismeri a mögötte rejlő lényeges folyamatokat
- A lényeglátást nagy mértékben segítheti egy rövid programozási alapképzés.
- Szövegszerkesztést jól megtanulni ennek hiányában is lehet, csak lényegesen hosszabb idő alatt.
- Adatbázis-kezelést viszont programozási és adatmodellezési alapismeretek nélkül nem lehet elfogadható módon elsajátítani.

E megállapításaimat elsősorban saját tapasztalataim igazolják.

A címben feltett kérdésre adott válaszuk: egyértelmű igen. Nem számítástechnika szakos hallgatóknak is érdemes megtanítani a programozás alapelemeit (input, output, adatok, műveletek, ciklus, elágazás és alprogramok). Kérdés csupán az, mikor. Például a szövegszerkesztési, táblázatkezelési vagy adatbázis-kezelési órák keretében, ha máskor nincs rá lehetőség.

Irodalom

- [1] Informatika és számítástechnika az oktatásban (INFO 92) konferencia, Békéscsaba, 1992

- [2] Timár L. : Elmélkedés a számítástechni oktatásával kapcsolatos elképzeléseinkről, *Informatika a Felsőoktatásban Konferencia*, Debrecen, 1993
- [3] Programozás - alapok nélkül? Új alaplap, 1995/5