

AZ INFORMATIKA OKTATÁSÁNAK MÓDSZERTANA

Szlávi Péter, Szlavi@ludens.elte.hu
Zsakó László, Zsako@ludens.elte.hu
ELTE TTK Informatikai Tanszékcsoport

Abstract

Informatics is a new subject in Hungarian primary and secondary education. It has been taught in large number of schools for a few years, although there is no uniform, standard curriculum. The reason for this is that a large number of informatics teachers were autodidacts in acquiring their knowledge of informatics. But there is a much more serious problem i.e. unlike mathematics or other disciplines there does not exist a unified methodology of informatics.

In this paper we are trying to summarize all the topics, tools and methods which have been worked out in Informatics Teacher Training at ELTE in the last few years.

1. Bevezetés

Az 'Informatika oktatása' című tantárgyat az Informatika tanárszakos hallgatók a 8-9. félévben hallgatják, heti két órában.

Az informatika oktatásának módszertana egy különleges tantárgy. Stílusában, módszereiben eltér az informatikai szaktárgyaktól, de más tárgyak (pl. matematika) szakmódszertanától is.

A tantárgy egyik feladata az informatikából közép- és általános iskolában tanítandó ismeret kör meghatározása, mind az alapképzésben, mind az informatikai szakképzésben. Az informatika világa nagyon gyorsan változik, s –szerencsére– a Nemzeti Alaptanterv sem köti meg túlságosan a változásokat követni akaró tanár kezét. Ez azonban azzal jár, hogy évről-évre újra kell gondolni a tanítandó anyag körét.

Másik –klasszikus módszertani– feladat az *informatika* egyes ismeretkörei *tanítási módszereinek megismerése*, valamint a részterületek áttekintése oktatási szempontból.

Harmadik célkitűzésünk a *tanári munka előkészítése*. Ennek érdekében hallgatóink a tárgy mindegyik félévében kötelezően tanítanak (jelenleg három kétórás gyakorlatot) a szak, illetve más tanárszakok alacsonyabb évfolyamain. A tanításban segítségükre van az adott csoport/tantárgy gyakorlatvezetője, aki a gyakorlóiskolai vezetőtanárokhoz hasonló módon támogatja, illetve értékeli a náluk oktató tanárjelölteket. Ugyancsak ide tartozik, hogy a létező tantervek megismerése mellett mindenkinek kötelező legalább egy tantervet és egy ehhez kapcsolódó tanmenetet készíteni. A tanári munka harmadik részeként részt vesznek programok, dolgozatok javításában (pl. a Nemes Tihamér OKSzTV kb. 300 programját első körben ők javítják).

A hagyományos órákon belül kétféle stílusú foglalkozás van: ismeretközlő *előadás*, ill. „*vezetett*” *eszmecsere*, amelyen nincsenek köbevésett igazságok, csak egy téma, amiről minden tanárjelöltnek kell, hogy véleménye legyen.

2. Témakörök

Az Informatika oktatásának módszertana tantárgy az ELTE Informatika tanárszakján az alábbi témakörökből áll:

- Az informatika/számítástechnika oktatásának célja, az informatika/számítástechnika oktatásának téveszméi és azok kritikái.
- Az informatika ismeretkörei, témák ismeretkörökbe és korcsoportokba sorolása.
- A Nemzeti Alaptanterv informatikai témakörei.
- Tananyag-felépítés, tantárgyfelépítés informatikából.
- Az Országos Képzési Jegyzék számítástechnikai szakmái.
- Speciális informatikai képzés lehetőségei.
- Informatika/számítástechnika tantervek és tankönyvek elemzése.
- Számítógép modellek.
- Programozási modellek, programozási alapfogalmak kialakítása. [1]
- Programozás módszertani variációk (Neumann-elvű, automataelvű, nondeterminisztikus, funkcionális, logikai, párhuzamos, objektumorientált, ...)
- A programozás, a programozási nyelvek, az alkalmazói rendszerek, az operációs rendszerek tanításának módszerei.
- Programozási nyelvek, alkalmazói rendszerek oktatási értékelésének szempontjai, nyelvek és rendszerek értékelése.
- Operációs rendszerek és felhasználói felületek.
- Magyarul vagy angolul?
- A programozás és a matematika – mi más a számítógépek világában.
- Dolgozatösszeállítás, dolgozatjavítás, hibatípusok, programok javítása, tesztelési stratégiák.
- Számítástechnikai feladatsorok összeállítása.
- Gépvásárlás, tanterem összeállítás, szoftver rendszer összeállítás.
- Informatika az iskolában tanórákon kívül – informatikai támogatású projektek.

3. A témakörök kibontása

A következőkben a fenti témakörök közül bemutatjuk azokat, amelyeket fontosságuk, ill. kevésbé elterjedtségük miatt feltétlenül ismertetendők tartunk.

3.1. Az informatika oktatásának célja, az informatika oktatás téveszméi és azok kritikái

A közismereti informatika tantárgy ismeretköre éles vitákat kelt az informatika tanárok körében is, de ennél sokkal élesebbet az informatikus szakemberek, valamint az informatikához laikus szülők esetében. A tanítandó területeket, azok fontosságát, arányait úgy próbáljuk megvilágítani, hogy megfogalmazzuk a hibás, szélsőséges nézeteket, majd kritikus szemmel megvitatjuk őket. Az elmúlt évben az alábbi téveszméket vettük nagyjító alá:

- Informatika tantárgy \equiv beszéljünk az informatikáról általában
- Informatikaoktatás \equiv szakképzés
- Informatikaoktatás \equiv csak alkalmazói oktatás
- = Alkalmazói oktatás \equiv alkalmazói program funkcióinak ismerete¹

¹ Az előbbi téveszme „al-téveszméje”.

- Informatikaoktatás \equiv csak programozásoktatás
= Programozásoktatás \equiv egy programozási nyelv oktatása
- Informatikaoktatás \equiv azonos súlyú ismeretkörök oktatása
- Informatikaoktatás \equiv az informatika önkényesen kiválasztott néhány részterülete
- Informatikaoktatás \equiv valamilyen informatikai eszköz teljes megtanulása
- Informatikaoktatás \equiv független ismeretkörök egymás utáni megtanítása
- Mindig a legújabb informatikai eszközt tanítsuk
- Az informatikatanár tudása kisebb, mint a tanulóé
- Az informatikaoktatáshoz nem kell Informatika tantárgy

Mindegyik téveszme tartalmaz természetesen valamilyen igazságelemet, de a végletes alkalmazásuk biztosan hibás. Az ezekről a téveszméről folytatott viták alapján alakul ki a hallgatókban, hogy az Informatika tantárgyban milyen ismeretköröket, s azokat milyen súllyal érdemes oktatni.

Ugyanennek a vitának egy másik eredménye az informatikaoktatás céljának meghatározása, amely az alábbiakban foglalható össze:

- korszerű alkalmazói készség kialakítása (a számítógépek, az informatikai kultúra lehetőségeit kihasználni tudó tanulók képzése)
- az algoritmikus gondolkodás fejlesztése (a matematikához hasonló gondolkodásfejlesztő szerep, amely az iskolában, s a hétköznapi életben is alapvető fontosságú);
- önálló munkára nevelés (a számítógép, mint a tanuló tevékenységére azonnal reagáló eszköz, lehetőséget teremt az egyéni ütemű tanulásra, a tehetségekkel való különleges foglalkozásra, ...);
- együttműködésre nevelés (nagyobb számítógépes feladatok megoldása megköveteli a csoportmunkát, feladatok részekre osztását, a másokkal való kapcsolattartást);
- termékelőállítási folyamat kialakítására, alkotó munkára nevelés (akár programot írunk a számítógéppel, akár szöveges dokumentumot vagy adatbázist, a végeredmény akkor is egy termék lesz, a készítés folyamatának, s a „termékségnek” minden egyes következményével együtt);
- az informatika és a társadalom kölcsönhatásának felismerése (az informatika rohamos fejlődése az egész társadalmat gyökeresen átalakítja, s ebben az állandóan változó világban csak az érezheti otthon magát, aki érti a változásokat, s azok mozgatóit);
- valamilyen közismereti tárgyhoz kapcsolódó témakörben a tanulók önállóan készítsenek el számítógépes anyagokat (program, dolgozat, előadás, bemutatás, ...).

3.2. Az informatika ismeretkörei, témák ismeretkörökre és korcsoportokra sorolása

Az informatikaoktatás céljaiból megállapítható, hogy az informatika ismeretek mely részét kell egy informatika tantárgyon belül megtanítani, s mely részét kell alkalmazni a többi tantárgyban, sőt az iskolai élet minden területén. Az informatika tantárgy tárgyalt ismeretkörei az alábbiak: [2]

- Adatmodellezés, algoritmizálás, programozás (az iskolai és a mindennapi életben lépten-nyomon algoritmusokat hajtunk végre, adatstruktúrákat –kérdőíveket, nyomtatványokat– töltünk ki, tevékenységsorozatot, információáramlási folyamatokat tervezünk, s ezt a világot az érti igazán, aki tisztában van ezen tevékenységek alapjaival).
- A programozás eszközei (azon nyelvi és egyéb eszközök ismerete tartozik ide, amelyek az algoritmusok, adatmodellek megvalósításához, kipróbáláshoz feltétlenül szükségesek).
- Alkalmazói feladatok megoldása számítógéppel (itt a mindennapokban felmerülő problémák informatikai eszközökkel való megoldhatóságával kell foglalkozni: ábrászerkesztés, szövegszerkesztés, táblázatkezelés, adatbáziskezelés, ...).
- Alkalmazói rendszerek kezelése (az alkalmazási ismeretektől el kell választani a gyorsan elavuló eszközök kezelésének képességét, bár a tanításuk természetesen párhuzamosan kell, hogy történjen).

- Gyakorlati problémák megoldása számítógéppel (ebben az ismeretkörben a felmerülő problémából kell kiindulni –pl. osztálykirándulást kell szervezni–, szervezési feladatként kell vele először foglalkozni, utána ki kell választani az egyes részfeladatokhoz tartozó eszközöket –nem feltétlenül mindegyik informatikai eszköz–, ha szükséges akkor új eszközt kell készíteni, ...).
- Informatikai eszközök működési elvei és alkalmazásuk (sokféle hardver és szoftver eszköz áll rendelkezésünkre, amelyek célszerű alkalmazását minden számítógéphasználónak el kell sajátítani).
- Informatika és társadalom (érdemes megismerni az informatika, mint a kultúra egy részterülete múltját, foglalkozni az informatika várható fejlődésével, valamint jelenlegi hatásával a társadalomra, adatbiztonsággal, adatvédelemmel, az informatikaalkalmazás etikai kérdéseivel).
- Az informatika matematikája (az informatikai ismeretek elsajátításához szükséges matematikai alapok a matematika tantárgyban vagy nem szerepelnek, vagy nem ott, ahol szükség lenne rájuk –ez így jó, pl. a matematika nem foglalkozik mátrixokkal, a táblázatkezelési ismeretek azonban e fogalom hétköznapi megfogalmazását szükségessé teszik–, tehát erről, valamint a matematika alkalmazásairól az informatika tantárgynak kell beszélni).

3.3. Tananyag-felépítés, tantárgyfelépítés informatikából

Ebben a témakörben meg kell vizsgálni az alapvető tananyag-felépítési stratégiákat, s azok alkalmazhatóságát az informatikaoktatásban: [2]

- lineáris (ismeretkörök valamilyen egymásutániségét feltételezi, különböző tanévekhez különböző ismeretköröket rendel),
- ciklikus, ismétlődő (azt feltételezi, hogy az informatika főbb ismeretkörei valamilyen szinten már a tantárgy első tanítási évében meg kell, hogy jelenjenek, s ezeket a következő tanévekben fokozatosan tovább mélyíti).

A másik meggondolandó a tantárgy tanéven belüli felépítésének kérdése. Ebből a szempontból háromféle típusú ismeretkört vizsgálunk: [2]

- az ismeretkör anyaga nagy blokkokban, folytonosan oktatható (pl. nincs értelme szövegszerkesztést úgy tanítani, hogy minden harmadik órán foglalkozunk vele),
- az ismeretkör anyaga egy- (esetleg két-) órás blokkokban tanítható, hosszabban biztosan nem szabad foglalkozni vele (elrettentő példaként szoktuk hozni azt az iskolát, ahol fél évig folyamatosan DOS ismereteket tanítottak),
- az ismeretkör anyaga maximum 5-10 perces órarészekben tanítható (pl. számítástechnika történetéről egy 45 perces óra már biztosan unalmas, de egy 5 perces mese –egy más típusú, de nehéz tananyag elsajátítása közben– nagyon élvezetes lehet).

3.4. Speciális informatikai képzés lehetőségei

A közoktatásban bátorítani kell gimnáziumok újszerű informatikai „szakképzéseit”, amelyek elsősorban a továbbtanulni nem szándékozókknak adnak olyan informatikai ismereteket, amelyek közvetlen munkába állásukat segíthetik. Ez a témakör kicsit más, mint a szakképzés informatikai szakmái, itt ugyanis nem arról van szó, hogy informatikusokat képezzünk, hanem arról, hogy az informatika eszközeit intenzíven használó nem informatikus szakembereket.

Az informatika oktatása tantárgyban kitalálunk, s tartalommal töltünk ki néhány olyat, amelyet akár a jelenlegi eszközparkkal is oktathatnának (sőt egyeseket jelenleg már oktatnak is) gimnáziumok. Az elmúlt tanévben tárgyalt „szakképesítések”: *informatikus titkárnő*, *kiadványszerkesztő*, *idegenforgalmi szervező*, *oktatástechnológus-információtechnológus* .)

Hasonlóképpen tárgyaljuk azt is, hogy a közismereti informatikán túl milyen speciális ismeretekkel kell rendelkezniük az egyes szakközépiskolák diákjainak. Ezen belül az alábbi nagy csoportokkal foglalkozunk:

- műszaki szakközépiskolák,
- közgazdasági-pénzügyi szakközépiskolák,
- agrár-élelmiszeripari szakközépiskolák,
- egészségügyi szakközépiskolák,
- kereskedelmi-vendéglátóipari szakközépiskolák.

3.5. A programozás, a programozási nyelvek, az alkalmazói rendszerek, az operációs rendszerek tanításának módszerei

Az informatika egyes közoktatásban megjelenő részterületei önmagukban is nagyok, s a tanításuk sem egyféleképpen képzelhető el. Az informatika tanárszakos hallgatóknak meg kell ismerkedniük mindazokkal a tanítási módszerekkel, amelyeket a közoktatásban (a nagyvilágban, valamint az egyetemi oktatásban rajtuk) már kipróbáltak, s több-kevesebb sikerrel alkalmaztak. Megbeszéljük az egyes módszerek előnyeit, hátrányait, s a hallgatóknak ezek alapján kell kiválasztaniuk azt a módszert, amelyet a bevezetőben említett gyakorló tanításban felhasználnak.

A tárgyalt programozástanítási módszerek :

- *Módszeres, algoritmusorientált* (a módszer a teljes programkészítési folyamatot egységében tárgyalja, de a programozási munka középpontjában az algoritmuskészítés áll, a programozási folyamat többi eleme ehhez csatlakoztatható),
- *Feladattípus-orientált* (egy konkrét feladatkörből, pl. grafika, szövegfeldolgozás, ... feladatsorokat épít, s a feladatsor megoldása közben vezeti be az éppen szükséges új programozási ismereteket),
- *Nyelvorientált* (programozási nyelvből kiinduló módszer, a nyelv elemeihez rendel programozási ismereteket – a közoktatásban klasszikus BASIC tankönyvek nagy része ilyen módszert alkalmaz),
- *Utasításorientált* (az előzőhöz hasonló, de nem egy konkrét programozási nyelvre, hanem egy általános nyelvtípusra építő módszer),
- *Matematikaorientált* (a matematika tantárgy igényeire építve vezeti be a programozási ismereteket, amelyhez egy matematikai diszciplína –pl. számelmélet– feladatigényeit veszi figyelembe),
- *Specifikációorientált* (a programozási folyamat leghangsúlyosabb részének a –lehetőleg– formális specifikálást tartja, a specifikációból automatikusan vezeti le az algoritmust, majd azt merev kódolási szabályok segítségével állítja elő a kódot),
- *Hardverorientált* (elképzelése szerint az algoritmikus ismeretek nem érthetőek magasszintű programnyelvi ismeretek nélkül; a programnyelvi ismeretek nem érthetőek assembly, ill. gépi kódú ismeretek nélkül; az assembly ismeretek nem érthetőek a processzor működésének ismerete nélkül; ...).

A tárgyalt programozási nyelv tanítási módszerek :

- *Utasításorientált* (a nyelvet utasítások halmazaként fogja fel, s a halmaz egyes elemeit tanítja valamilyen sorrendben),
- *Segédeszközként alkalmazás* (a programozásoktatás, illetve adatbáziskezelésnél az adatbázisoktatás szempontjait tekinti elsődlegesnek, s a szükséges mértékben vezeti be hozzá a nyelvi eszközöket),
- *Feladattípus-orientált* (a programozástanítási módszereknél tárgyalattal azonos),
- *Nyelvorientált* (a nyelvet strukturált egységnek tekinti, a nyelv logikáját állítja előtérbe, s a szükséges mértékben vezeti be a nyelv konkrét elemeit),
- *Mintapélda alapján* (a nyelvet mintapéldák elemzésén keresztül mutatja be).

Az alkalmazói rendszerek tanítási módszerei rövidebb múltra tekinthetnek vissza, s így kevésbé kidolgozottak, kipróbáltak, mint a fentiek voltak. Közülük az alábbiakkal foglalkozunk:

- *Problémaorientált* (fokozatosan bővülő feladatsorokhoz választ alkalmazói eszközt, illetve bővíti a használandó eszközökből használt fogalmakat, funkciókat),

- *Menüorientált* (az alkalmazói ismereteket az alkalmazói programon keresztül tanítja, méghozzá annak menüpontjai szerint),
- *Funkcióorientált* (meghatározza az alkalmazói rendszer általános funkcióit, pl. szövegszerkesztőnél: beírás, javítás, nyomtatás, formázás, ...; majd a funkciókhoz kapcsolja a konkrét ismereteket),
- *Fogalomorientált* (meghatározza az alkalmazói rendszer által használt fogalmakat, pl. táblázatkezelőnél: cella, sor, oszlop, blokk, lap, ...; majd ezen fogalmakhoz kapcsolódó funkciókat tárgyal),
- *Absztrakt eszköz orientált* (definiálja egyre bővülő absztrakt eszközök sorozatát, pl. szövegszerkesztőnél: hagyományos írógép, elektronikus írógép javítási funkcióval, elektronikus írógép sorformázási funkcióval, ...; majd az egyes szintekhez szükséges ismereteket tárgyalja),

Az operációs rendszerek tanításának módszereivel azért kell foglalkozni, mert jelenleg ez az, amit az informatikatanárok a lehető legrosszabbul csinálnak. Itt igazából nem is módszerekről, hanem csupán alapelvekről lehet beszélni:

- az operációs rendszer parancsok (menüpontok, ikonok) halmaza, s ezt a maga teljességében tanítsuk meg,
- az operációs rendszer egy feladatkör ellátására szolgáló program, s a tanuló számára más célú feladatmegoldáshoz feltétlenül szükséges funkcióit kell megismerni.

3.6. Programozási nyelvek és alkalmazói rendszerek értékelése

Néhány általános megjegyzéssel kezdjük, amelyből a szoftverek *oktatási céljai* és az oktatás ezen szoftverekkel szemben támasztott *elvárásai* derülnek ki. Majd rátérünk a programozási nyelvek értékelésére.

A programozási nyelvek céljait így látjuk:

- az algoritmikus gondolkodás tanításának gyakorlati eszköze
- egy feladatmegoldás valamely adott témakörben (pl. egy számítógépi szimulációs program elkészítése valamely más szakterületi probléma modellezésére)
- egy nyelvtípus megismer(tet)ése
- (bizonyos körülmények között) professzionális programozói ismeretek adása

Legelőször is megbeszéljük, milyen szempontok játszhatnak szerepet a nyelvek megítélésében. Ezeket szoktuk említeni:

- nyelvi egyszerűség
- tipikusság
- a nyelvi környezet egyszerűsége
- használhatóság
- megbízhatóság
- fejlesztői környezet
- biztonságosság
- bonyolultság

Miután a szempontokat tisztáztuk a hallgatók által, a 'Programozási nyelvek' tantárgy [3] keretében megismert programozási nyelveket mindegyikükkel értékeltetjük (egy formanyomtatvány kitöltésével). A téma lezárása a „közös vélemény” kialakításával történik. Idén a nyelvhalmaz a következőket tartalmazta:²

- Turbo Pascal
- Comenius LOGO
- Turbo PROLOG [4]
- Assembly
- Borland C++ [5]
- ELAN [6]
- LCN LOGO [7]
- Quick BASIC [8]
- TopSpeed Modula-2 [9]

Az alkalmazói rendszerek jellemzése az előzőhöz hasonló „szerkezetben” történik. Azaz a tanítási célok és jellemzési szempontok megfogalmazása után a gyakorlatuk során előforduló konkrét rendszerek önálló, majd közös értékelése. (A gyakorlatokra való apellálást részben az 'Informatikai rendszerek' tantárgy, részben olyan más tantárgyak teszik lehetővé, amelyek „mellékesen” használatnak velük egy-egy alkalmazói rendszert.) Célok:

- mindennapi alkalmazói ismeretek nyújtása
- egy rendszertípus megismerése
- (bizonyos körülmények között) professzionális alkalmazói ismeretek adása

Értékelési szempontok:

- egyszerűség
- vizualitás
- teljesség
- rugalmasság
- megbízhatóság

A vizsgált konkrét rendszerek idén az alábbiak voltak:

- táblázatkezelők kategóriájából – Quattro Pro, Excel
- szövegszerkesztők kategóriájából – Norton Editor, ChiWriter, Word, WordPerfect, WinWord, TEX

3.7. Számítógépmodellek

Többfajta megközelítést is megbeszélünk. Az egyikük [10] lényege, hogy a némileg a kitűzött célnál többre is vállalkozik. Ugyanis nem elégszik meg a számítógép működésének modellezésével, hanem a „számítógép-evolúciót” végigkövetve képet ad a számítógép históriájáról, és sok-sok informatikai kérdést vet föl, s mutat be egy lehetséges választ. Érdekessége: egy többfelvonásos dráma keretei között zajló eseménysor vázolja a gondolatokat. Ez lehetővé teszi, hogy mozgalmasan, élvezetesen, hitelesen „adja elő” a tanár tanítványai közreműködésével. A felvonásokra tördelés arra is fölhasználható, hogy önállóan, időben

² Meg kell jegyeznünk: évek óta ez nem sokat változott.

egymástól akár távolra helyezve, más témákkal felváltva, mintegy a „komoly” témák közötti pihenésül alkalmazza a tanár.

Az egyes felvonások „mondanivalója” dióhéjban:

1. felvonás: megismerjük az alapszituációt és a szereplőket (Lóti Futit és társait) – a számítógép felépítése és legegyszerűbb működése
2. felvonás: új szereplő lép a színre – a számítógép működéséről továbbiak
3. felvonás: Lóti Futiek tanulmányosságáról – gépi kód, assembly, magasszintű programozási nyelv
4. felvonás: a tudomány és a népszámlálás kapcsolatáról – alapvető perifériák, operációs rendszer
5. felvonás: a népszerűség ára – a párhuzamos végrehajtás (gazdálkodás erőforrásokkal)
6. felvonás: Lóti Futi osztódik, sőt még telefont is kap – a párhuzamosságról továbbiakat (több-processzoros rendszerek, hálózatok)
7. felvonás: a mesterséges agy közeledése a természeteshez – 5. generációs számítógépek, mes terséges intelligencia

3.8. Operációs rendszerek és felhasználói felületek

A téma felvetését egy kérdőív kitöltésével kezdjük, ami a későbbi megbeszélés vezérfonalaként is szolgál. Az alábbiakban néhány tipikus kérdést idézünk a kérdőívből.

- Miért vannak (lennének jók) az iskolákban számítógépek?
- Miért érdekes a számítógépek programfelületének milyensége?
- Milyen fajta felületkategóriákat különböztessünk meg?
- Kell egyáltalán a „csupasz” operációs rendszer oktatása?
- Mikor használ az ember, ha alkalmazó, operációs rendszer ismereteket, s miket?
- Mikor használ az ember, ha programozó, operációs rendszer ismereteket, s miket?
- Milyen témákat (fogalmakat) célszerű tisztázni minimálisan az operációs rendszerrel kapcsolatosan?

Itt egy táblázatot töltünk ki a táblán a kérdőívek alapján. A szempontokat a hallgatók találják ki, mi csak egy-két induló ötlettel járulunk hozzá.

<i>Szempont</i>	<i>Előfordulás-szám</i>
berendezések	
file-név konvenciók	
könyvtárszerkezet	
file-csoport kijelölések	
...	

1. táblázat

- Mire lehet szükség egy „fél-grafikus” operációs rendszer kiterjesztés ismereteiből?
- Mik az előnyei egy grafikus felületnek?
- Milyen minimális tudnivalókat érdemes egy grafikus felülettel kapcsolatban megtanítani?
- Milyen sorrendben, s hogyan lehet oktatni e három „felületet”?

Talán a kérdőívből is kitalálható, hogy a három felhasználói felület: a „hagyományos” operációs rendszer (pl. DOS, UNIX) „parancsorientált” felülete, egy Norton Commander-szerű, „félgrafikus”, „billentyűorientált” felülete valamint egy Windows-féle sokablakos, ikon-orientált felület.

3.9. Magyarul vagy angolul?

A nyelvi témát fontosnak tartjuk fölvetni, hiszen a tanárjelölt lépten-nyomon találkozni fog ennek problematikájával. A probléma fölvetésével kezdjük. Imígyen:

Az informatika társadalmasodása miatt mindenki találkozik fogalmaival, eszközeivel. Így szükségszerűen az informatika beszivárog a *kultúrába*, részévé válik annak. Ami a kultúrához tartozik, azzal az *oktatásnak* foglalkoznia kell: el kell helyeznie az ember mindennapjaiban, meg kell tanítani bánni eszközeivel, érteni fogalmait, módszereit...

Az informatika *angol nyelvterület*en (Anglia, USA) alakult ki kb. az 1940-60-as évek között, itt körvonalazódtak az elmélet és gyakorlat alapjai. Tehát *terminológia angol-szász* eredetű. A rohamos fejlődés is elsősorban itt következett be, s tart azóta is. Más nemzetek legfeljebb lépést igyekeznek tartani ezzel a piaci-tudományos-gazdasági életet (tehát gyakorlatilag az élet minden területét) behálózó diszciplína (sőt ma már inkább diszciplínákat helyes mondani) terminológiájával, formálásával. Általában leginkább csak a befogadására marad idő, de arra is eléggé ellentmondásosan. Minden nemzet másként áll hozzá a „befogadáshoz”. Vannak olyan „önérzetes”, nagy nemzetek, amelyek nem hajolnak meg az angol befolyás előtt. S saját utat próbálnak járni: saját nemzeti terminológiát kialakítani, s használni. Ilyenek a franciák és a németek. Vannak olyan népek, amelyek nyelvéről nem mondható el, hogy világnyelv volna, ezeknél ez a törekvés nemigen működik.

Ilyen gyors és világméretű *kultúrsokk* még nem érte az emberiséget, ezért nincs még recept ennek biztonságos nyelvi, tudati és kulturális identitást megőrző túlélésre. (A robbanásszerű változások miatt nincs idő megvárni, hogy a nyelv a „saját, megszokott” tempójában feldolgozza, s majd sajátjaként befogadja az új nyelvi, fogalmi jelenségeket. Márpedig elzárkózni nem lehet e hatástól, mivel ezzel az élet más dimenzióiban szükségképp lemaradásra ítélné saját magát az adott nemzet.)

Ezt követően a magyar nyelvet korábban ért hatások nyelvi emlékei közül sorolunk föl néhányat az érdekesség kedvéért. Majd párhuzamot vonunk nyelvi szemszögből az általános kultúra fejlődése és az informatika fejlődése között, a következtetések levonása érdekében. (A kultúra utolsó fél évezrednek az informatika fél évszázadát szembesítjük.)

	<i>Az általános kultúra története</i>	
Középkor	irodalom, tudomány = latin-görög kultúra	csak kevesek, az uralkodó réteg privilégiuma, az oktatás magántanárok által ...

Reneszánsz = az 1. nemzeti öntudatra ébredés („internacionális esemény”!)	nemzeti nyelvű írásos kultúra, ami első sorban profán témákat ölel föl a művészetekben , bár a nemzeti nyelvű bibliafordítások révén az egyházi kultúrában is megkezdődik a magyarosodás	iskolák jönnek létre, nemzeti nyelven is folyik az oktatás: többek hozzáférnek a kultúra eredményeihez (a szegény deákok is!)
Felvilágosodás, nyelvújítás	a beszélt és irodalmi írott nyelv közeledése, és általánossá válása az élet legkülönbözőbb területein: tudományban, művészetekben, gazdasági, társasági életben	az oktatásban nemzeti nyelven történik a tanítás: minden író-olvasó a kultúra

2. táblázat

	<i>Az informatikai kultúra története</i>	
Informatikai középkor	programozói, mérnöki tudomány = angol nyelvű tudományos kultúra	szűk, szakmai grémium birtoka az informatikai tudás, „saját nevelésű folytatók”, egyszemélyes iskolák
Informatikai reneszánsz	„önálló”, nemzeti, tudományos kutatások : (fél)magyar nyelvű terminológia az informatikában	az informatika felsőfokú oktatásbeli bevezetése: szakember-utánpótlás céljából
Informatikai felvilágosodás	az informatikai eszközök bevonulása a hétköznapokba	az informatika oktatása bevonul a közoktatásba – természetesen magyarul

3. táblázat

Az elkövetkezőkben az informatikai szókinccs magyarosításának két fontos szempontjára hívjuk föl a figyelmet: az *önérzetesség* re és a *praktikusságra*. Önérzetesség: valóban érdemes szép nyelvünkkel törődni, de azt a praktikusság talaján szabad csak végezni. Az első szemponthoz néhány nyelvünket dicsérő példát idézünk a nyelvészeti szakirodalomból. A másodikhoz, jól és kevésbé ismert, bevált és elsorvadt nyelvújítási szavainkból sorjázunk egy tucattal, amit megtetézünk pár frissen született informatikai szókéssérettel, szórejtellyel, sőt –megítélésünk szerinti– szóborzalommal.

Eközben a hallgatóság mozgósítására és a praktikusság szempontjának érzékeltetésére egy tesztet osztunk ki két csoportnak. Mindkét csoport tesztlapján informatikai fogalmaink közül található néhány. Egyiken eredeti angol, a másikon újkeletű magyar változata. Ki kell találni a párját. Elgondolkodtató a két csoport „eredményessége”.

4. Következtetés

A tantárgy –hitünk szerint– nélkülözhetetlen információkkal és tapasztalatokkal képes ellátni az informatika szakos tanárjelölt hallgatókat. Ennek azonban „kemény” feltételei vannak. A fentiekből kiderült, hogy igen érzékenyen kell reagálnia –legalábbis ma még– az oktatónak az adott év tematikájának összeállításánál, hogy az informatika és az oktatásügy gyors mozgásával lépést tudjon tartani, ami fontos az élő, tartalmas tananyaghoz. A gyors és pontos reagálás megköveteli viszont, hogy az oktatás szereplői „jelen legyenek” más, az informatikát érintő területen is. Aktív oktatói, kutatói tevékenységet folytassanak a

programozás módszertanban, a programozási nyelvek és alkalmazói rendszerek világában, s –nem kevésbé fontos, hogy– az informatika népszerűsítéséből, az informatikai közoktatás formálásából is igyekezzenek részt vállalni.

Irodalomjegyzék

- [1] Szlávi P.-Zsakó L.: Programozási modellek: Hétköznapi algoritmusok. Inspiráció 2, No. 1, 21-23, No. 2, 24-25, 1994.
- [2] Zsakó L.: Az informatika ismeretkörei. (μLógia 30) ISzE - ELTE TTK Általános Számítástudományi Tanszék, 1995
- [3] Szlávi P.- Zsakó L.: Programozási nyelvek - Alapfogalmak. (μLógia 22) ELTE TTK Általános Számítástudományi Tanszék, 1994
- [4] Makány Gy.: Programozási nyelvek - Prologika. (μLógia 7) ELTE TTK Általános Számítástudományi Tanszék, 1989
- [5] Illés Z.: Programozási nyelvek - C++. (μLógia 23) ELTE TTK Általános Számítástudományi Tanszék, 1993
- [6] Szlávi P.- Zsakó L.- Temesvári T.: Programozási nyelvek - ELAN. (μLógia 24) ELTE TTK Általános Számítástudományi Tanszék, 1994
- [7] Ökrös L.: Programozási nyelvek - LCN LOGO. (μLógia 25) ELTE TTK Általános Számítástudományi Tanszék, 1993
- [8] Hack F.: Programozási nyelvek - BASIC. (μLógia 28) ELTE TTK Általános Számítástudományi Tanszék, 1996
- [9] Kozics S.: A Modula-2 programozási nyelv. ELTE TTK Általános Számítástudományi Tanszék, 1992
- [10] Szlávi P.: A számítógépről népszerűsítő stílusban. (μLógia 5) ELTE TTK Általános Számítástudományi Tanszék, 1990