



HungaroLogo

Első magyar Logo konferencia

1994. október 25-27.

szervezők:

**Neumann János Számítógéptudományi Társaság
Informatika Számítástechnika Tanárok Egyesülete**

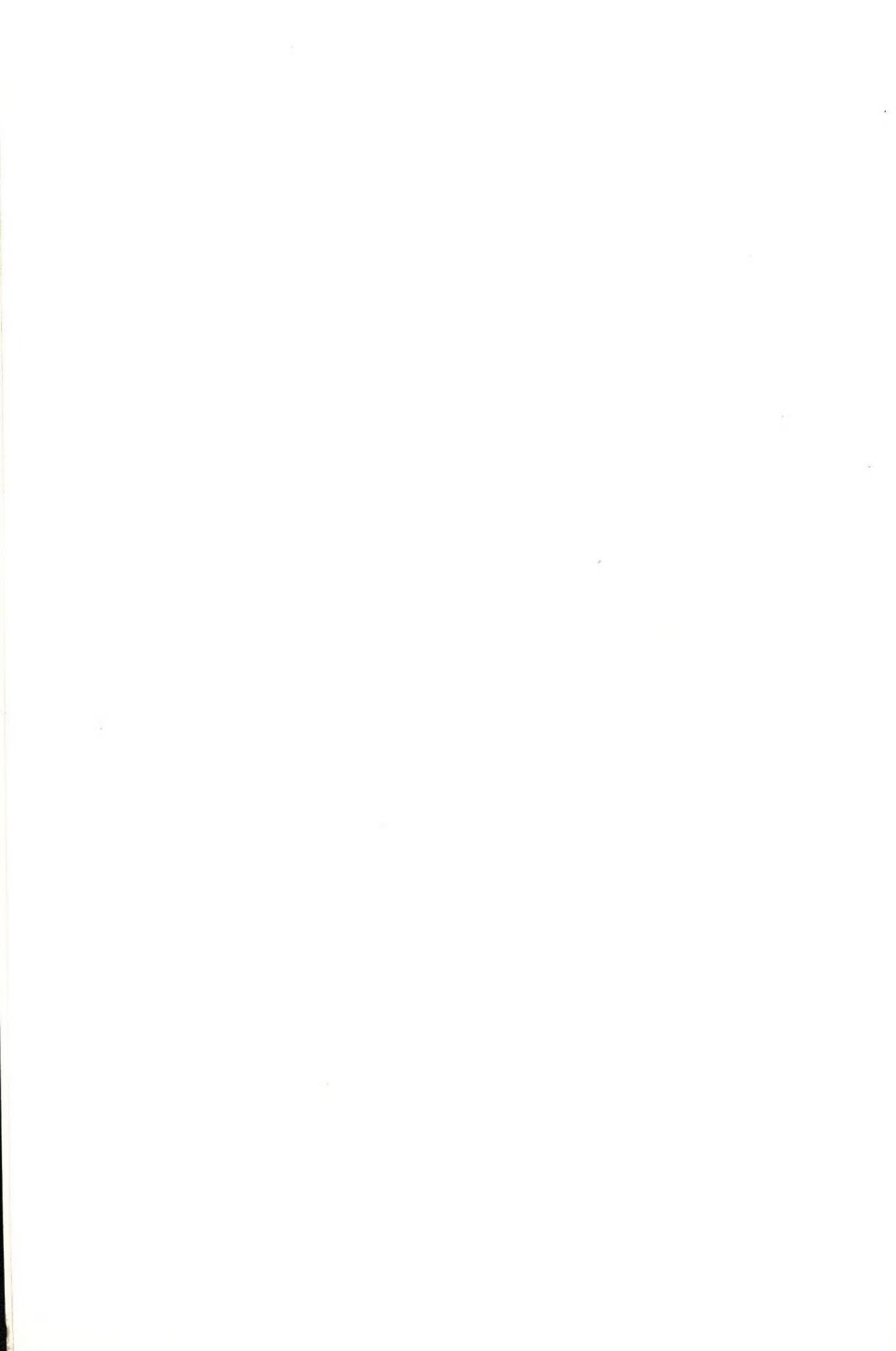
Helyszínek:

Budapesti Tanítóképző Főiskola
ELTE Általános Technika Tanszék
Alternatív Közgazdasági Gimnázium
Bocskai Úti Általános Iskola
Érdi Úti Általános Iskola

ITA/331



iTA/331





HungaroLogo

Első magyar Logo konferencia

1994. október 25-27.

szervezők:

**Neumann János Számítógéptudományi Társaság
Informatika Számítástechnika Tanárok Egyesülete**

Helyszínek:

Budapesti Tanítóképző Főiskola
ELTE Általános Technika Tanszék
Alternatív Közgazdasági Gimnázium
Bocskai Úti Általános Iskola
Érdi Úti Általános Iskola



A konferencián elhangzó, írásban előzetesen beküldött tanulmányok gyűjteménye.
Szerkesztette: Farkas Károly

Kiadja: a Neumann János Számítógéptudományi Társaság

A címlapon a "robmalac" Farkas Ivett ComLogo-val készült munkája.

Tartalom:

Bevezetés

Forgács Tamás: Áhá élmények a LOGO-ban

Kőrösné dr. Mikis Márta: LOGO az informatika tantervekben: miért, mikor, hogyan?

Könczöl Tamás: Animációk a LogoWriter-ben

Bedő Ferenc: Munkafüzetek a Logo programozási nyelv tanulásához

Garádi Istvánné: Az 5. osztályos geometria tanítása a LogoWriter program segítségével

Kőrösi Lajos: Vegyipari berendezés rendszerbiztonsági elemzése hibafa segítségével élelmiszeripari technikumokban Logo programozási nyelven

Dr. Nyéki Lajos: A Logo nyelv oktatása a műszaki pedagógusképzésben a Széchenyi István Főiskolán

Baharevné Nádudvari Katalin: A számítógép alkalmazásának lehetőségei a beszéd-nyelvtanulási zavarok diagnózisára épített terápiában a hallássérült előkészítő tanulóinknál...

Törtely Éva: Az informatika játékos tanítása Pilisvörösváron

Farkas Károly: Logo nyelv és logo-pedagógia

Kalas Ivan, Blaho Andrej: Not All Birds Are Turtles. Másodközlés
Az előadás elhangzott: UNESCO Workshop Teacher Education and Communication and Information Technologies, Enschede, 1994.

A NAT informatika részének megvalósításához az ISZE által készített javaslat szerint is a Logo lehet az egyik elsőként alkalmazott programnyelv. (Az angliai tanterv egyedül ezt a nyelvet ajánlja az óvodától az egyetemig.)

A magyar általános és középiskolákban évek óta terjed a Logo nyelv, az általános iskolákban legelterjedtebb változat a LogoWriter.

Hazánkban is működnek a Logo nyelvvel foglalkozó kutató csoportok. A Neumann János Számítógéptudományi Társaság Logo Szakosztályának és az ISZE vezetőinek is egyik célja ezen kutatások összefogása.

A Logo nyelvvel, a logo-pedagógiával foglalkozó nemzetközi kutatómunkában évek óta magyar szakemberek is részt veszünk. A második EUROLOGO konferencián Gentben három, a harmadikon Pármában egy, a negyediken Athénben négy magyar szakember vett részt. Az elhangzott előadásaink és a konferenciák megrendezése között működő tudományos bizottsági munkánk elismerése is az, hogy az 1997-es hatodik konferenciát Budapesten rendezzük meg. (Az ötödik 1995-ben Birmingham-ben lesz.)

Ezen előzményekre alapozva, és ezen célok elősegítésére is ez év októberében a Neumann Társaság és az ISZE megrendezi a

HUNGAROLOGO

konferenciát, a első hazai országos, a Logo nyelvvel és a logo-pedagógiával foglalkozó tanácskozást.

A konferencia elnöke: Vámos Tibor akadémikus

Az előadók között az ELTE és a tanítóképző főiskolák oktatói, a Logo nyelv szakemberei, közép és általános iskolai tanárok, tanítók és gyógypedagógusok lesznek. Kiváló szakemberek jelentkeztek Erdélyből és a Felvidékről is. Az előadások négy szekcióban hangoznak el:

Logo mint programnyelv
Logo a középiskolában
Logo az általános iskolában
Logo-pedagógia és óvoda

A szekciók vezetői: Turcsányiné Szabó Márta ELTE
Forgács Tamás ÁKG
Farkas Károly BTF
Kőrösné Mikis Márta OKI

A harmadik napon a Logo és a TechnoMir interfészrendszer együttes működtetését mutatjuk be. Helyszín az ELTE Általános Technika Tanszéke.
Bevezető előadást Szücs Ervin tanszékvezető tart.

Forgács Tamás: Áhá élmények LOGO-ban
1. Hungarologo Konferencia, Budapest 1994 október

1. Bevezetés

Áhá élménynek nevezzük, ha valamilyen váratlan dologra bukkanunk, felismerünk valamit, ott ahol nem is reméltük, rájövünk valami új és váratlan összefüggésre. stb. Seymour Papert *Észregés* című könyvében hatásos ötletnek (powerful idea) nevezi azokat a gondolatokat, amik valami új felfedezésére vezetnek pl. a LOGO segítségével.

Valóban, aki bármilyen szinten és környezetben foglalkozik LOGO-val, annak nagy esélye van arra, hogy Áhá élményekkel találkozók.

A Játékos Informatika Oktatás egyenesen arra törekszik, hogy a gyerekeknek minél több Áhá élményben legyen részük.

Aki aktív módon - tehát az olvasottakat ki- próbálva és továbbgondolva - olvassa Abelson és di Sessa *Turtle Geometry* című könyvét, elég sűrűn részesülhet Áhá élményben. Csak egyet említenék meg, a polygon fogalmának meglepő és kissé váratlan kiterjesztését.

En most olyan Áhá élményekről szeretnék beszámolni, amelyek a LOGO-val, mint programozási nyelvvel, egész pontosan a LOGO-val, mint a LISP listakezelő nyelv egyik dialektusával kapcsolatosak. Beszámolok arról is, hogy mi inspirálta ezeket az élményeket. A témákon még dolgozom, ezért főleg az ötlet, azaz az Áhá élmény szempontjából beszélek róluk.

2. Miért nincs a LOGO-ban "tömb" adattípus?

A válasz kézenfekvő, nincs rá szükség, hiszen a listastruktúra - ami a LOGO-ban alapvető - egy komplexebb struktúra, ami magában foglalja a tömbstruktúrát is.

A dolog persze nem ilyen egyszerű.

Mert az igaz, hogy egy tömbstruktúra mindig visszavezethető egy listastruktúrára, - mint ahogy például egy **for** ciklus is rekurzív függvényhívásra -, de a cél éppen az, hogy a már megszokott régi formát használjam. A megoldás kézenfekvő, írnom kell olyan procedúrákat, amelyek a tömbhíváshoz hasonlóan (tehát pl. indexeket használva) kezelik a listát, ott, ahol ez lehetséges. Ez nem látszik nehéz feladatnak:

```
TO arrayread :tomb :i :j
```

```
  op iten :i item :j thing :tomb
```

```
END
```

ez kiolvassa egy tömb *i,j* indexű elemét

Valamivel bonyolultabb, de viszonylag egyszerű a

```
TO arraywrite :tomb :i :j procedura megírása is
```

Az első gond az, hogy a tömböknél feltesszük, hogy azonos típusu elemek vannak, és minden tömb sor azonos hosszú kell, hogy legyen. Ha komolyan vesszük, hogy tömbökkel dolgozunk, akkor ezeket a tulajdonságokat illik ellenőrizni. Ez sem komoly probléma, bár itt azért már van néhány kérdés, amit nem triviális eldönteni, például az, hogy hogyan döntjük el, ill hogyan deklaráljuk, hogy egy tömbemnek mi a típusa.

Nos mi lenne, ha azt mondanánk, hogy nem ellenőrizzük ezeket a kikötéseket, és éppen ezért be sem tartjuk őket, ha ezt úgy látjuk célszerűnek. Persze most már másról van szó, most már nem tömbökkel dolgozunk, hanem listákkal csak éppen tömbszerűen kezeljük őket, azaz a lista elemeit indexeléssel kapjuk meg. Persze ekkor előfordulhat, hogy egy `arrayread :i :j` nem talál értéket, mivel ilyen elem már nincs is a tömbben, ilyenkor mondjuk a "NIL sztringet adjá vissza. Értelmet kapnak olyan információs függvények, mint pl a

dimension "tömb [3 2], ami visszadja, hogy max hány index fűzhető még ehhez a tömbkifejezéshez, aholis a negatív válasz is értelmezhető. Könnyű belátni, hogy például háromszögmátrixok nagyon jól ábrázolhatók itt. Az indexeléses litakezelésnek egyik legnagyobb előnye lehet, hogy ha felakarjuk jegyezni egy lista adott helyét, elég csak az indexeket feljegyezni, vagy az is, hogy algoritmusunkat arraa bázírozzuk, hogy az indexek számolhatók.

Nyilvánvaló, hogy ezt a módszert jól használhatjuk nagyon hiányos táblázatok kezelésére. Itt gondot okozhat, ha a hiány nem a sor végén van, akkor egy hézagpotló nullelemet kell alkalmazni. Ezen a problémán segíthet a "property list" fogalmának általánosítása, de ez még csak egy powerful idea (talán) és különben sem fér be ennek a dolgozatnak a kereteibe.

3. LOGO-ban mint tudjuk könnyű olyan procedurát írni, aminek a paramétere lehet LOGO utasítássorozat is. De akkor le kellene tudni ellenőrizni, hogy az így beadott utasítássorozat korrekt LOGO - e, vagy legalábbis ki kéne zárni a nyilvánvalóan nem azokat.

Vagyis az a kérdés, hogy milyen nehéz LOGO-ban valami szintaxis ellenőrzésfélét vagy ahhoz hasonlókat írni.

Ha csak olyan LOGO utasítássorozatra számítunk, ami kizárólag primitíveket tartalmaz viszonylag egyszerű a dolgunk. A .primitives utasítás visszadja egy listában az összes primitívet. A LOGO utasítássorozatban csak szét kell választanunk a primitíveket és a konstansokat, (ez utóbbiak formailag könnyen felismerhetők) és akkor egy **member? .primitives** kiszűrheti az "elgépelt" primitíveket. Ez az eljárás általában elég jól kiszűri azokat az inkorrekt LOGO utasításokat amik elgépelés útján jöttek létre.

Ha a hívások komplettégét is ellenőrizni akarjuk, akkor szükségünk van egy összetettebb primitív táblázatra is, ami feltünteti, hogy egy adott primitívnek hány paramétere van.

Újabb problémába akkor ütközünk, ha meg akarunk bírkozni az olyan nem tiszta LOGO procedurákkal, amelyeket infix operátorként használunk, mint pl. a + a - vagy a < jel, és ez maga után vonja a zárójel szükségességét is. Ezen egy lexikális preprocessor segíthet, ami "Polish notation"- ná alakítja a zárójeles kifejezéseket. Ez egy jól ismert algoritmus, semmi elvi nehézséget nem okoz a megírása.

Újabb komoly probléma jön, ha nemcsak a primitíveket akarjuk ellenőrizni, hanem a felhasználó által írt procedurákat is. Nyilvánvaló, hogy ehhez a primitív táblázatunkba fel kell vennünk az új procedurákat is, ami egyáltalán nem triviális feladat. Írhatunk pl egy procedurát, amit minden procedura elején kötelező meghívni, és ami a hívó procedúra jellemzőit viszi fel egy listára.

De itt most érdemes egy kicsit megállni. Miért is fontos nekem, hogy LOGO utasításokat ellenőrizzek, hiszen ezt megteszi nekem a LOGO maga.

Az viszont már több szempontból is érdekes lehet, ha olyan feladatokat oldok vagy oldatok meg, amihez köteles vagyok egy szűkített LOGO-t használni, vagy még sokkal jobb ötlet, hogy primitíveket egyáltalán nem engedek meg, csak egy általam létrehozott procedúrahalmazt használhat a hallgató, egy olyat ami komplett nyelvi rendszernek tekinthető. Egy ilyen saját nyelv szintaktikus ellenőrzését a fenti módszerek alkalmazásával könnyedén tudok írni.

Egy példa az utóbbira. Olyan teknőcgeometriát akartunk használni, amiben lehetséges volt, hogy a teknőc útját egy akármilyen előre definiált téglalap belsejébe szorítsuk. Ehhez viszont teljesen új teknőcgeometriát kellett írni. Eme teknőcgeometria utasításait nem célszerű a régiekkel keverni, Ezért egy szintaktikus szűrő itt nagyon jól jön.

Befejezésül megemlítenék egy olyan Áhá élményt, amit Shafee Give'on (Israel) cikke (4. Eurolog Konferencia Athen 1993) inspirált. Shafee bevezet a végtelen sorozatokra egy csomó operátort, amik segítségével a végtelen sorozatokra egyenleteket lehet felírni. Ezekkel az egyenletekkel azután jól ismert végtelen sorozatokat lehet előállítani. Ezeket az operátorokat azután meg lehet LOGO metafüggvényekkel valósítani. Továbbá végtelen sorozatokat reprezentáló procedurákat is lehet írni (függvény a sorozat n-ik elemének kiszámítására, procedura a sorozat első n elemének kinyomtatására, stb). Ezen LOGO procedurák azután nagyon alkalmasak arra, hogy különböző nem triviális végtelen sorozatok tulajdonságait tanulmányozzuk, ill. ezeket szemléletes módon tanítsuk.

Egy példa a jól ismert Fibonacci sorozatra.

Legyen S egy végtelen sorozat, és jelentse BUTHEAD S az S sorozatot az első eleme nélkül. Akkor S egy Fibonacci sorozat, ha

$S + \text{BUTHEAD } S = \text{BUTHEAD BUTHEAD } S$

Az én ötletem az, hogy vezessük be a fenti technikát a differenciálható függvényekre. Függvényoperátor legyen pl a DIFF deriváltoperátor és az INT integráloperátor.

Ekkor például a $\text{DIFF } f = \text{INT } f$ vagy a $\text{DIFF } f = f$ egyenlet meghatározza az exponenciális függvényt. Vagy a $f / \text{DIFF } f = c * x$ a hatványfüggvényt határozza meg. Nyilvánvaló, ha jó függvényreprezentációt lehet találni, akkor ez a rendszer a különféle differenciálegyenletek tanulmányozására lesz nagyon alkalmas.

Eme Áhá élményem feldolgozásának még csak az elején tartok, egy későbbi cikkben szeretnék beszámolni az esetleges eredményekről.

Budapest, 1994 október 10.

Irodalom

Shafee Give'on, Y

Catch Infinity by its Tail. Infinite Sequences in LOGO, Revisited
Proceedings of the fourth European Conference, Athen 1993

Brian Harvey
Computer Sciences Logo Style
MIT Press 1985

LOGO az informatika tantervekben: miért, mikor, hogyan?

Kőrösné dr. Mikis Márta:

Országos Közoktatási Intézet

A 80-as évek végétől az informatika, mint új műveltségterület fokozatosan kivívta helyét az iskola falai között is, sokszor egy régebbi, számítástechnikának nevezett tantárgy utódként. Tanításának tartalmát döntően befolyásolták és úgyszólván determinálták az iskola tárgyi és személyi feltételei. Míg kezdetben a kazettás home-computerek a számítógépes játékok és a BASIC-programozás alkalmazását ösztönözték, a PC-k megjelenése és elterjedése már újfajta informatikai szemléletmód kialakulását hívta életre. Az egyszerűbb és a profi színvonalú, látványos és könnyen kezelhető alkalmazói programok lehetővé teszik, hogy kicsik és nagyok, amatőrök és szakértők egyaránt személyes céljaikat, ötleteiket, gondolataikat megvalósíthassák a gép segítségével.

A PC-k világában nem "ragadhatunk le" egy FOR-TO-NEXT ciklus szintaxisánál, ehelyett **levelet írunk, meghívót rajzolunk, zenélünk, adatokat kezelünk, robotot vezérlünk vagy éppen animációt varázsolunk** a képernyőre. Mindezt igen egyszerűen, különösebb "programozói szakismeret" nélkül, hála az egyre többet tudó és minél könnyebben kezelhető hardvernek és szoftvernek. Az alkotásra, a kreatív gondolatok kibontakoztatására már kisgyermekkorban is igen alkalmas az a **pedagógiai környezet**, amelyet a LOGO

segítségével valósíthatunk meg. Az elmúlt tanévekben számos iskolakísérlet és egyedi tanterv megpróbálkozott a **LOGO-programnyelv** tanításával, sőt, pedagógiájának alkalmazásával. A tapasztalatok igen pozitívak voltak. Kérdés viszont, hogy az átalakulófélben levő magyar oktatási rendszer és a központi tantervek hogyan viszonyulnak majd a LOGO-pedagógiához? Egyedi iskolai próbálkozás lesz vagy szélesebb népszerűségnek örvendhet a jövőben?

Mint tudjuk, a **NAT**-ok (Nemzeti Alaptantervek) eddigi összes változata kiemelten, önálló műveltségi blokkként ajánlja az informatika tanítását. A NAT mellé elkészülő követelményrendszer korcsoportonként rögzíti azokat a legfontosabb tudnivalókat, amelyeket az adott korosztálytól elvárhatunk. A tananyag, a tudás közvetítéséhez vezető utat az iskoláknak maguknak kell megkeresniük. Helyi tanterveket kell készíteni, megválasztani a szükséges taneszközöket a tankönyvtől kezdve a szemléltető táblán át egészen a számítógépig. Mindehhez, de főképp **az újdonságként megjelenő informatika tantárgy helyi tanterveihez** ötleteket kell adnunk. Hogyan illeszthető a LOGO-környezet az új tantárgy tantervébe? Mi indokolja alkalmazását? Hány éves kortól ajánlott a megjelenése és milyen óratervi feltételekkel?

A LOGO oktatási megjelenése hazánkban már nem újkeletű. Hivatalos formában az OPI által 1989-ben megjelentetett, a minisztérium által jóváhagyott **Technika-Informatika modulterv** lehetővé tette alkalmazását az általános képzésben, a technika tantárgy órakeretén belül. Számos iskola élt is ezzel a lehetőséggel. Napjainkban viszont, amikor egyre inkább nyilvánvalóvá válik, hogy az általános műveltség szerves részét képezik az informatikai alapismeretek, nem

lehet csupán egy iskola vagy pedagógus személyes ügye annak eldöntése, hogy tanítson-e informatikát vagy sem, illetve azon belül milyen témakörökkel foglalkozzon.

A helyes szemléletmód kialakításában és a helyi tantervek elkészítésében igen hasznos segítséget jelent az a szakanyag, amelyet az Informatika-Számítástechnika Tanárok Egyesülete készített közel 120 pedagógus bevonásával "**A Nemzeti Alaptanterv Informatika Kerettanterve**" címmel. Ez a tanterv mátrix elrendezésben is szemlélteti a tananyagot témakörökre és korcsoportokra bontva. A legegyszerűbb informatikai alapfogalmaktól kezdve az alkalmazói programrendszereken át az informatika társadalmi vonatkozásáig 9 nagy témakört tartalmaz az anyag. Igen helyes, hogy nem feledkezik meg a kisgyermekek korcsoportjáról sem (6-10 év), bár ekkor még más tárgyakba elbújtatva és inkább játékos formában jelennek meg az alapvető ismeretek.

A LOGO iránt elhivatottságot érző pedagógusok nagy örömeire szolgál, hogy a szerzők elfogadták a "**paperti gondolatokat**". Bár példaanyagként szolgál, de számos helyen találkozunk a LOGO ajánlatával: a programozás eszközeinél éppúgy, mint az alkalmazói ismereteknél. Mindez a LOGO sokoldalúságából fakad, hiszen **lehetőséget ad a legkülönfélébb informatikai alapfeladatok realizálására**: grafikus és karakteres elemek képernyőn történő elhelyezésére, a kép egyszerű átalakítására (mozgatás, átszínezés), kinyomtatására, lemezen történő tárolására stb, mindezekre rendkívül látványos formában és éppen ezért rendkívül motiváló hatással.

Az ajánlást átlapozva választ kapunk a feltett kérdésekre. A LOGO bármely korcsoportnál - a megfelelő szinten közvetítve - eredményesen alkalmazható és alkalmazandó. Ezt igazolják a **nemzetközi tapasztalatok** is: tanulmányutak, illetve konferenciák (például a legutóbbi EUROLOGO) győznek meg bennünket arról, hogy a LOGO-környezet a fejlett országok óvodáiban éppúgy megjelenik, mint az általánosan képző alap- és középfokú intézményekben. Logikus és kreatív gondolkodásra nevelése valamint látványossága és egyszerű kezelhetősége miatt külföldön a fogyatékos gyermekek képzésében is sokhelyütt alkalmazzák.

Az 1994/95-ös tanévben megkezdődik a Nemzeti Alaptanterv követelmény- valamint vizsgarendszerének kidolgozása. A központi, egységes és merev tantervek megszűnnek, szabad kezett adva a pedagógusnak és az iskolának a választásra, a követelmények elsajátításához vezető saját út kidolgozására. Az informatika helyet kap az új tantárgyi rendszerben. Azon kell fáradoznunk, hogy a számítógép és ezáltal **az informatikai kultúra minél hatékonyabban jelenjen meg az oktatásban**. Hiszen a most oktatott gyermekeinknek már a jövő évezred erősen informatizált társadalmában kell élniük, alkotniuk. Ehhez pedig az iskolának kell megadnia a felkészítést.

Animációk **LogoWriter**-ben

Előadó:

Könczöl Tamás
informatika tanár

GALGAMÁCSAI ÁLTALÁNOS ISKOLA

TALENTO ALAPÍTVÁNY ISKOLÁJA

1. Bemutakozás: - A BUDAPESTI TANÍTÓKÉPZŐ FŐISKOLÁT és a debreceni KOSSUTH LAJOS TUDOMÁNYEGYETEM informatika tanári szakát végeztem el. Öt éve tanítok a GALGAMÁCSAI ÁLTALÁNOS ISKOLÁBAN; ettől az évtől pedig a TALENTO ALAPÍTVÁNY iskolájában informatikát. Egyik alapító tagja voltam a Játékos Informatika Oktatás szakmai társaságnak. Az alsó tagozatos informatikai képzésben az elsődleges célom a játékos képesség és személyiségfejlesztés, a tanulást segítő tevékenységek támogatása.

2. A LOGO és az eljárás-orientált gondolatvilág

A LOGO-t elsődlegesen nem mint programnyelvet, hanem mint eszközt használom fel a kítűzött pedagógiai céljaim elérésére.

Kezdetben minimális programozási ismeretet nyújtok, az utasításokat sem kell megtanulni kívülről, mert szerintem itt nem a lexikális tudás a mérvadó, hanem a kreatív alkotás.

Nem a programozási részt emelem ki, hanem a teknőcöt; a LOGO gondolati fogódzóját, a vele való rajzolást, felszabadítva ezzel a számítógépes környezet adta kötöttséget, utat adva az alkotóképes tudás kibontakozásának.

A LOGO támogatja a moduláris felépítést, eljárásokban gondolkodik.

A problémák részfeladatokra bontásával kétféle programozási módszer alkalmazható vele:

- a) TOP-BOTTOM, vagyis felülről lefelé haladó;
- b) BOTTOM-UP, vagyis alulról felfelé haladó.

Ezek a módszerek alkalmasak az összetettebb alakzatok elemekre bontására, vagyis a nekik megfelelő eljárások megírására, majd ezekből a kész rajz összeállítására, vagyis az eljárások összefűzésére (ház probléma).

Ez a módszer preventív jellege miatt is jól használható. Az egyes eljárásokat külön teszteljük, hibáit javítjuk, majd csak ezután kapcsoljuk össze őket. Így a hibalehetőségek nagy részét is kikerüljük.

3. A komplex problémamegoldás egy lehetséges változata

Az animációk készítése komplex feladat, mivel a következőkből áll:

- háttértervezés-készítés technóc primitívekkel ;
- mozgó objektumok tervezése-készítése / SHAPES / alakzatokkal ;
- hangeffektek, zene készítése témától függően ;
- az előző részek összefűzése homogén egésszé - a vezérlő eljárás megírása ;

4. A munka menete

Első lépésben tanári problémafelvetésre - tanulói ötletek alapján az elérendő cél felvázolása történjen meg. Ez lehetőleg ne legyen azonos az egész osztály számára, hanem a gyermekek-gyermekcsoportok érdeklődési körének feleljen meg.

Ha ez megtörtént, bontsuk fel a munkát témacsoportokra / lásd fent / , biztosítva ezzel a modularitást; készítsük el a munkamenet algoritmusát. Ezt elegendő szóban megtenni.

Ne válasszunk túl bonyolult feladatokat, mert az elhúzódó munka esetleg nem biztosít folyamatos motivációt.

A munkát kezdjük a háttér elkészítésével.

Ezt is lehet tovább bontani, az egyes részeket külön eljárásban megírni / pl.: ház, út, hegy, / , majd összefűzni ezeket.

A háttér nem csak vonalas rajz lehet, ki is lehet tölteni a rajzokat különböző színekkel.

Második lépésben a mozgó objektumokat készítsük el / pl.: autó, emberke, helikopter, ... / , vagy a menüből SHAPES pontot aktiválva, vagy a rajzlapról a SHAPES utasítást begépelve. Itt fontos megemlíteni, hogy a beágyazott objektumokat ne változtassuk meg, van még sok hely újak készítésére. Ha mégis azok közül akarunk egyet módosítani, másoljuk át / F3-F4 / egy üres helyre, és ott módosítsuk.

Az egyes mozdulatok alakzatai egymást kövessék, hogy könnyen tudjuk őket hívni sorszámukkal.

Harmadik lépésben készíthetünk hozzá témától függően hangeffekteket is. Pl.: ha egy autót irányítok, és „nekimegy valaminek ” , szólaljon meg egy felfutó effekt.

Zenét, dallamot írni szintén megpróbálhatunk, és ezt akár „játékunk” több helyén is megszólaltathatjuk.

A legnehezebb feladat az utolsó lépés, megírni a vezérlő eljárást.

Ez kezdődjön a háttérrajz behívásával, majd folytatódjon az alakzatok megjelenítésével, a kellő helyen.

Készíthetünk címlapot is, mint egy filmben, a készítőik neveivel.

A nehézséget az alakzatokat mozgató eljárás megírása jelenti. Itt - szükség esetén - figyelni kell a rajz határvonalait / COLORUNDER / , be kell építeni az animáció végét jelentő kilépési lehetőséget.

Az effekteket általában bizonyos koordináták elérésekor szólaltatjuk meg, a komplett dallamokat pedig a program kezdetén és/vagy a végén.

Segítséget lehetőleg egyénileg, az adott probléma körüljárásával, az azt redukáló irányított kérdésekkel, és ne a megoldás megadásával adjunk.

5. Gyermekmunkák bemutatása

Második, harmadik, negyedik osztályos munkák bemutatása számítógépen.

6. Értékelés

Véleményem szerint az animáció készítés a LOGO-pedagógia azon területe, ahol a legkomplexebb a feladat, ezért ezek készítése a kisgyermek képességeit igen sokoldalúan fejleszti.

Mint vázoltam, a problémafelvetéstől a megvalósításig igen sokféle típusú feladat adódik, ezek elkészítése, áttekintése és legfőképp - integrálása átfogó fejlesztést jelent, miközben legnagyobb a motiváció / játékot írhat /, ezért észre sem veszi a feladat - életkorához viszonyítottan - nehézségét.

Saját tapasztalataim azt mutatják, hogy az alapok - teknőc primitívek használata, eljárások írása-egybefűzése - elsajátítása utáni legjobb hatásfokú munkavégzés LOGO-ban az animációk készítésével érhető el.

Ajánlom tehát minden kollégámnak, aki még nem alkalmazta, próbálja ki, a siker nem marad el.

Köszönöm figyelmüket: *Köncsöl Tamás*

Munkafüzetek a Logo programozási nyelv tanulásához

Bedő Ferenc

Községi Általános Iskola

Pusztaszentlászló

Kossuth u. 112.

8896

Hazánkban a mikroszámítógépes korszak "hőskorában", az 1985-86-os években, amikor legtöbbször a BASIC programozási nyelv oktatásának szükségességében hittek, már felmerült a gondolat, hogy az általános iskolás korú tanulókkal inkább a Logo nyelvet kellene megismertetni. Az ilyen irányú vélemények még meglehetősen bátortalanul hangoztak el, hiszen a megvalósításnak még nem voltak meg a technikai feltételei. Az általános iskolákból még hiányoztak azok a nagyobb memóriakapacitású számítógépek amelyeken komolyabb Logo fordítóprogram is futtatható lett volna. Ez az akadály a C64 számítógép használatával elhárulni látszott, az utóbbi években pedig amikor már ezen iskolaformában is megjelentek az IBM kompatibilis PC-k és az Apple Macintosh gépek, hardver szempontból nincs akadálya a Logo nyelvvel való foglalkozásnak.

Az utóbbi években, különösen Seymour Papert: Mindstorms c. könyvének hazai kiadása után hazánkban megnőtt az érdeklődés a nyelv tanítása iránt.

Az Informatika-Számítástechnika Tanárok Egyesületének Informatika NAT javaslata, 10-12 éves tanulók esetében a Programozás alapfogalmai és a Programozás eszközei c. témakörök tanítása során ajánlja a Logo nyelvvel való foglalkozást.

Véleményem szerint a Logo nyelv tanításának van jövője oktatásunkban, hiszen, ha - a hazai pedagógiai gyakorlathoz igazodva - csak egy "megtanítandó tananyagként" kezeljük, akkor is helyet követelhetne magának az általános iskolás tantervekben. De ne tekintsük csak annak, hiszen jól tudjuk, hogy működése során egy hatékony "számítógépes környezetként" fog funkcionálni. A Logo környezet, "Logo-műhely" a "számítógépes környezetek" csak egy előfordulási lehetősége, mégis az oktatás számos területén segítheti munkánkat. Azt, hogy egy ilyen jellegű számítógépes műhely, a magyar gyerekek körében milyen készségeket, képességeket fejleszt, mely területeken segít, milyen tananyagok elsajátítását könnyíti meg, annak teljes felderítése hazánkban még az elkövetkezendő évek pedagógiai kutatásainak témája lesz.

A Logo-val kapcsolatos szakirodalom néhány éve öröndetesen bővül, mégis tanítása során a tanulók által használható segédeszköz még nem közismert.

E hiány enyhítésére vállalkozó eszközök egyike szeretne lenni az általam készített két munkafüzet, amely a 10-13 éves korosztály Logo programozási nyelv tanulását segítheti.

A tankönyv forma helyett előnyösebbnek tartom a munkafüzet formát, hiszen a Logo-t én nem tekintem "megtanítandó tananyagként". A Logo igazi haszna a vele való foglalkozásban, az észrevétlen tanulásban rejlik, és ennek inkább megfelel a munkafüzet, mint tanulási segédlet.

A Logo nyelvet 10-12 éves gyerekekkel már akkor szerettem volna megismertetni amikor az általános iskolákban jobbra még csak C16 és C+4 számítógépek voltak. Mivel nem tetszettek az akkor e gépeken futtatható "Mini Logo" változatok, magam készítettem BASIC nyelven egy egyszerű fordítót C+4 számítógépre. E programmal kezdtem el első kísérleteimet, valamint ehhez készítettem el a jelen munkafüzet első változatát.

Jó néhány éves kísérletezés és szakirodalom tanulmányozás során alakítottam ki egy megtanítható és megtanítandó ismeretanyagot. Nem törekedtem arra, hogy a teljes Logo nyelvet elhelyezzem a tananyagban, hiszen a cél nem a nyelv utasításkészletének maradéktalan megtanítása. Véleményem szerint a felső tagozatban feladatunk csak azon elemek ismertetése kell hogy legyen, amelyek segítségével a gyerekek fejlődését biztosító "Logo környezet" megteremthető, az alapvető algoritmusok, ill. programozási ismeretek bemutathatók.

Erre az általam kidolgozott tematikára építettem fel a két munkafüzetet, amelyek megírásával nem volt céлом önálló tanulást lehetővé tevő segédanyag készítése. A foglalkozások során szükség lesz a tanári magyarázatokra és segítségre is. Mindkét részben 32-32 lecke segítségével dolgozhatók fel az alábbi témakörök.

Kezdő rész

Teknőc-grafika
Egyszerű ciklusok
Eljárások készítése
Eljárások kezelése
Alapműveletek
Építőelemek
Programépítés alulról felfelé

Haladó rész

Hibakezelés
Összetett ciklusok
Programépítés felülről lefelé
Változók, műveletek változókkal
Véletlenek
Szavak, listák
Adatbeolvasás
Döntések, feltételek
Rekurzió, rekurzív ábrák
Koordináták

A témakörök közül többet is kiemelt jelentőséggel kezeltem, ezek gyakorlására több leckét is szántam. Ilyenek a Teknőc-grafika, építőelemek, valamint a programozástechnikai fejezetek.

A feladatok előtt, a leckék elején mindig a tanult elmélet egy nagyon rövid emlékeztető összefoglalása található. Ezen elméleti összefoglalók a tanár kollégáknak is segítséget adhatnak egy fejezet ismeretanyagával kapcsolatban. Az elméleti összefoglalót követi a leckénkénti 4-8 feladat.

A leckéken kívül megtalálható lesz a munkafüzetekben az alapszavak gyűjteménye, hibaüzenetek táblázata, valamint egy oldalon, kivágható teknőc figura és egyszerű szögmérő is a kezdeti lépések segítésére.

Igyekeztem többféle típusú feladatot szerkeszteni. Vannak amelyek egy egyszerű tevékenység számítógépen való elvégzésére utasítanak, végrehajtásuknak nem lesz nyoma a munkafüzetben.

A legtöbb feladat Logo programsor, programsorok, eljárások megalkotását tűzi ki célul. Mindezek kezdetben hiányosan találhatók meg, és a tanulóknak kell kiegészíteniük a sorokat, egyre több és több önálló munkára adva lehetőséget. A sorokban a kiegészítéseknek kipontozott hely van fenntartva. Olykor a teljes programot, eljárást a tanulónak kell megalkotni, ilyenkor a kipontozott hely elmaradhat.

A Teknőc-grafika tanulása során sokszor, a megadott ábrát megrajzoló program, eljárás elkészítése a feladat. Ennek fordítottja is előfordul, ekkor a megadott programsorok számítógépen való kipróbálása előtt kell elkészíteni a tanulóknak a várható ábrát.

A Logo-val való foglalkozásra az általános iskolai oktatásban több területén is látok lehetőséget. Az informatika tantárgy, ami az elkövetkezendő években nagyon sok iskolában bevezetésre kerül, a legvalószínűbb területe lesz a Logo nyelv tanításának. Ezen kívül a "Logo-zás" elképzelhető számítástechnika szakkörön, matematika szakkörön, képességfejlesztő szakkörön, technika, vagy matematika tantárgyon belül. Mindezek előfordulhatnak az általános iskola 5.-8. osztályában, a nyolc osztályos gimnáziumok 1.-4. osztályában, vagy a hat osztályos gimnáziumok 1.-2. osztályában. A munkafüzeteket igyekeztem úgy kialakítani, hogy használhatók legyenek az említett területeken.

Lehetnek a foglalkozások heti egy órásk és az éves tananyag zömét a Logo-val való foglalkozás teszi ki, ekkor első tanévben a Kezdő a másik tanévben a Haladó munkafüzet használható.

Lehetséges akkor is a használat, ha a tantárgy, vagy a szakkör heti egy vagy kettő órás, és a Logo az egy éves tevékenységnek egy kisebb részét képezi. Ilyenek lehetnek az informatika tantárgy, számítástechnika szakköri munka, matematika szakkör, képességfejlesztő foglalkozások.

Iskolánkban a Logo nyelvvel való foglalkozás az Informatika tantárgy 5. és 6. osztályos tananyagának képezi részét, a tantárgy évi 36 órájából kb. egyharmad részt elfoglalva.

A munkafüzetnek már volt egy szerényebb elődje. Azt a Commodore 64 számítógépen működő GEOS munkaprogrammal állítottam össze, ennek átalakításával született, a C64 gépen futó Terrapin Logo-t bemutató segédanyag.

A jelenlegi változatot WINWORD 2.0 kiadványszerkesztővel készítettem el, és alakítottam ki úgy, hogy azt, az IBM kompatibilis és Apple Macintosh gépeken egyaránt ismert, jelenleg talán legnépszerűbb Logo nyelvjárás, a LogoWriter alkalmazása esetén tudják használni a tanulók.

A továbbiakban (igény esetén) még lehetségesnek tartom, az átalakítást más Logo változatoknak megfelelő formába is.

A közelmúlt tapasztalatai alapján örömmel állapítottam meg, hogy a tanárkollégák körében jelentősen megnőtt az érdeklődés a Logo nyelv iránt. Ez bátorított arra, hogy a munkafüzetekhez kapcsolódóan, tanári kézikönyv összeállításába kezdjek. Jelenleg e munkának még nagyon az elején tartok, de a munkafüzetekhez, a fentebb említett alkalmazási területek némelyikében alkalmazható tanmeneteket már jelenleg is tudok nyújtani.

Bízom abban, hogy munkám segíteni tudja a Logo nyelvnek a hazai iskolai oktatásban való elismertetését és elterjedését.

**AZ 5. OSZTÁLYOS GEOMETRIA TANÍTÁSA A
LOGOWRITER PROGRAM SEGÍTSÉGÉVEL**

Garádi Istvánné

2.sz. Általános Iskola

Szigetvár, Szt. István ltp.

Iskolánkban az informatika oktatás már az alsó tagozat 2. osztályában elkezdődik. A kisgyerekekhez közelálló, játékos tanulásal ismerkednek meg a számítógéphasználat alapjaival. Ezért 5. osztályos korukra már jól tudják, hogyan kell a menüből kiválasztani és előhívni azt a munkaterületet, ahol éppen gyakorlást kívánnak végezni. (Betűismeret, szóalkotás, helyesírás, idegennyelv, dal-tanulás, számfogalom, alpműveletek, stb.) Nagy előnye a LOGO-nak, hogy a különböző tantárgyakhoz kapcsolódó ismeretek elsajátítását a megértés, az alkalmazás és az alkotás szintjén is képes biztosítani. Egyetlen szoftverrel (pl: LOGOWR) is megoldható tehát a sokoldalú képességfejlesztés, ha kell a korrepetálás vagy a tehetség-gondozás bizonyos része. Az kell hozzá, hogy a tanítók és a tanárok megismerkedjenek előnyös tulajdonságaival és a kész eljárások használatával. Nálunk egy iskolán belüli logo- tanfolyam inspirálja a kollegákat a különböző eljárások készítésére és így tudjuk a tanulást-tanítást számítógéppel segíteni. Más feltétel, pl: a gépter-mi háttér megteremtése ennél sokkal nehezebb. Tervezzük, hogy a matematika témaköreihez folyamatosan kidolgozzuk azokat az eljárásokat, amelyek használata hozzásegíti a tanulókat - a teknőc irányítása közben - saját tapasztalaton és ezek gondolati feldolgozása-n keresztül a tiszta fogalmak, a törvényszerűségek biztos ismereteihez.

Az átfogóbb, szilárdabb alapozás és a geometria látásmód fejlesztése érdekében az 5. osztályos geometria tanítása a síkban való pontos tájékozódás kialakításával kezdődik.

A derékszögű koordináta-rendszer.

Az eddigi gyakorlat helyett, (a padban ülő gyerekek, emeletes ház ablakai, stb.) a tanulók rögtön a saját képernyőjükön helyezhetik el a teknőcöt vagy a maga által készített ábrákat. Ezt általában két lépéssel tudták megoldani, (előre vagy hátra és jobbra vagy balra). Örömeikre szolgál és ezért megjegyzik, hogy egyetlen utasítással is megoldható a pontos hely kijelölése, csak a kezdő állapothoz képest előjeles számokkal kell megoldani az ellentétes mozgatási irányokat. SETPOS [140 -80]. Kész eljárásokat használunk a koordináta tengelyek, a négyzetrács, a pontrács kirajzolására. Ezek egyetlen szó beírásával bármikor előhívhatók, vagy eltüntethetők. Tankönyvi, vagy más érdekes feladatok megoldására is motiváltak a gyerekek. Ez segíti a hatékony gyakoroltatást, a koordináták helyes használatának rögzülését.

Itt nyílik lehetőség az egyenes, félegyenes, szakasz fogalmának tisztázására, a koordinátákkal megadott sokszögek megkülönböztetésére. A kerület-terület fogalom kialakítására ill. mélyítésére is sor kerülhet.

A téglalapok kerületének és területének kiszámítására olyan gyakorló eljárást használunk, ahol a teknőc színes tollal körbejárja a kerületet ill. befesti a területet. Dallammal jutalmazza, ha jól számítjuk ki a megfelelő mennyiségeket. Hiba esetén megmutatja a helyes számítási módot és újabb adatokkal másik téglalapot rajzol.

A szög fogalmához a tapasztalat szintjén, a teknőc forgatásával és közben a sík súroltatásával, mint tevékenységgel juthatunk el. A véletlen elfordulások becslése, lekérdezése (HEADING), adott nagyságú szögek rajzoltatása segíti a szögfajták közötti eligazodást, a nagyság szerinti összehasonlítást. Minden szög balra és jobbraforogatással is kirajzoltatható, bármilyen helyzetű is a nyugvó szár. A szögmérést a képernyőre bármikor behívható skálával ellátott korong (szögmérő) segíti.

A metsző, egymást merőlegesen metsző és párhuzamos egyenesek témakörénél a fő hangsúly a felismerésen van. A gyakorlást a pillanatok alatt újabb és újabb egyenesekkel telerajzolt képernyő biztosítja. A távolság fogalmának kialakításához felhasználjuk a COLORUNDER és DISTANCE parancsokat. A merőleges egyenes állítása az egyenesen kívüli pontból is megoldható egy előre elkészített eljárással. A sokszögek átlójával, magasságával kapcsolatos feladatok egyaránt segítik a gyakorlást és a tiszta fogalom kialakítását is.

Négyszögek megkülönböztetése és rajzoltatása tulajdonságaik alapján

A gyerekek párhuzamos egyenespár közé, tetszőleges oldallal és szöggel rajzolnak négyszögeket. A szimmetria és a megfelelő szögek összegének vizsgálatával eljutnak oda, hogy a különböző trapézok, paralelogrammák, téglalapok jellemző tulajdonságait felismerjék. Diszkutálásra is van lehetőség. A deltoidok rajzoltatásánál, a

szimmetriatengely-átló és az oldalak nagyságának összehasonlítása-
kor felvetődik a konvexitás kérdése.

A terület fogalmának kiterjesztése trapézokra, deltoidokra

Négyzetté vagy téplalappá való átdarabolásukra szemléletes eljárások írhatók. A képernyőn a szétvágott darabok mozgatása, majd összerakása leköti és ráirányítja a gyerekek figyelmét a területek egyenlőségére.

A számítógépes feldolgozás feltételezi, hogy vele párhuzamosan függetlenül, feladatlapot használjanak a tanulók. Így alakulhat ki a jelölésmódok, a vonalzó; a körző, a szögmérő helyes használata.

Rajzoláshoz, méréshez, szerkesztéshez a feladatlapok készülhetnek a LOGO eljárások alapján, nyomtatással.

**VEGYIPARI BERENDEZÉS RENDSZERBIZTONSÁGI ELEMZÉSE
HIBAFÁ SEGÍTSÉGÉVEL
ÉLELMISZERIPARI TECHNIKUMOKBAN
LOGO PROGRAMOZÁSI NYELVEN**

**KÖRÖSI LAJOS
CSVMTKF**

Bonyolult, összetett ipari berendezések hibalehatóságainak megvizsgálása, balesetek, esetleges katasztrófák megelőzése szempontjából törvényszerűen fontos.

A programban alkalmazott módszer a RENDSZERBIZTONSÁGI ELEMZÉSEK Hibafa elemzések MSZ-09-960615-87 szabványra támaszkodik. A program alkalmazása hasznos lehet élelmiszeripari technikumokban, folyamatok, rendszerek megismertetése, hibáinak feltárása céljából. Az előadás pontjai:

1/ BEVEZETÉS

2/ RENDSZERBIZTONSÁGI ELEMZÉSEK

Hibafa elemzések MSZ-09-960615-87

3/ A HŰTŐRENDSZER, MINT VEGYIPARI RENDSZER

a/ Hűtőkör bemutatása /1. sz. melléklet/

b/ Egy hűteterem recirkulációs hűtőrendszerének bemutatása a terem elektromos és kalórikus kapcsolási rajza alapján. /2.sz. melléklet/

4/ A HIBÁK MINT ESEMÉNYEK, A FELTÉTELRENDSZER FELÉPÍTÉSÉHEZ.

5/ A HIBAFÁ FELÉPÍTÉSE, ELEMZÉSE /3. sz. melléklet/

Az előadáshoz kapcsolódó program Logo nyelven a szemléltethető folyamatokat kíséri.

Bonyolult, összetett ipari berendezések hibalehetőségeinek megvizsgálása, balesetek, esetleges katasztrófák megelőzése szempontjából törvényszerűen fontos.

A programban alkalmazott módszer a **RENDSZERBIZTONSÁGI ELEMZÉSEK** Hibafa elemzések MSZ-09-960615-87 szabványra támaszkodik. A program alkalmazása hasznos lehet élelmiszeripari technikumokban, folyamatok, rendszerek megismertetése, hibáinak feltárása céljából. Az előadás pontjai:

1/ BEVEZETÉS

2/ RENDSZERBIZTONSÁGI ELEMZÉSEK

Hibafa elemzések MSZ-09-960615-87

3/ A HÜTŐRENDSZER, MINT VEGYIPARI RENDSZER

a/ Hűtőkör bemutatása /1. sz. melléklet/

b/ Egy hűteterem recirkulációs hűtőrendszerének bemutatása a terem elektromos és kalórikus kapcsolási rajza alapján. /2.sz. melléklet/

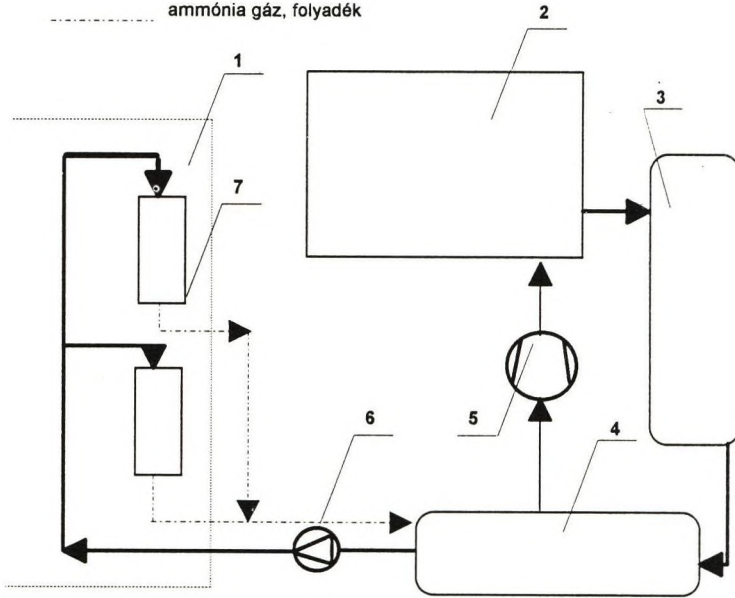
4/ A HIBÁK MINT ESEMÉNYEK, A FELTÉTELRENDSZER FELÉPÍTÉSÉHEZ.

5/ A HIBAFEA FELÉPÍTÉSE, ELEMZÉSE /3. sz. melléklet/

Az előadáshoz kapcsolódó program Logo nyelven a szemléltethető folyamatkat kíséri.

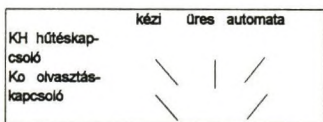
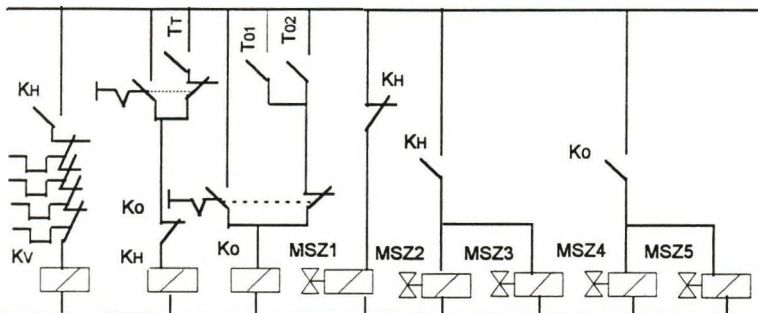
HŰTŐKÖRFOLYAM

- ammónia folyadék
—— ammónia gáz
- - - ammónia gáz, folyadék



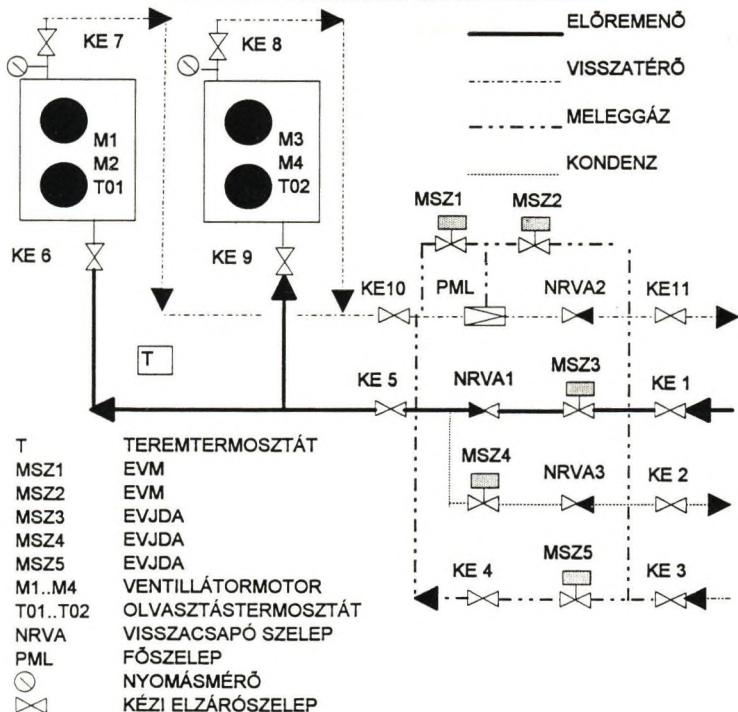
- 1- hűtött helyiség
2- kondenzátor
3 - folyadékgyűjtő edény
4 - folyadékleválasztó edény
5 - kompresszor
6 - ammónia szivattyú
7 - elpárolgató

2. sz. melléklet
A HŰTŐTEREM ELEKTROMOS KAPCSOLÁSI RAJZA



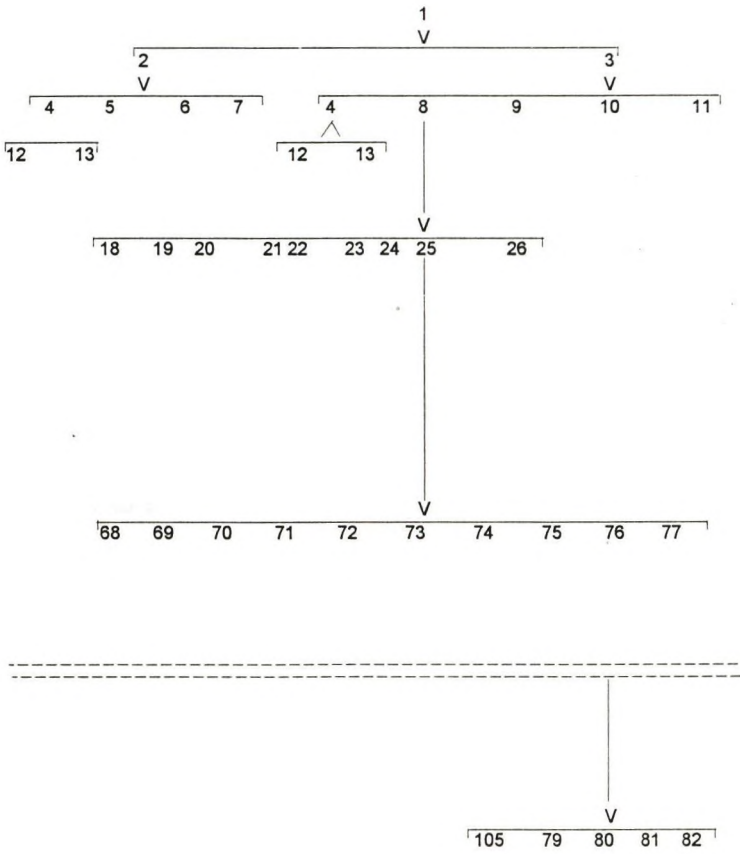
- TT - teremtermostát
 T01..T02 - olvasztástermostát
 MSZ1..MSZ5 - mágnesszelepek
 Kv - ventilátormotor

A HŰTŐTEREM KALÓRIKUS KAPCSOLÁSI RAJZA



3. sz. melléklet

HIBAFA /RÉSZLET/

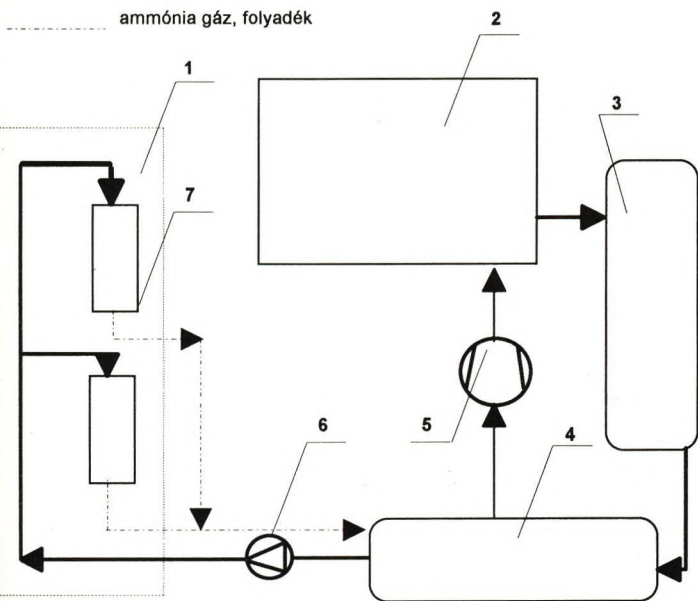


HŰTŐKÖRFOLYAM

———— ammónia folyadék

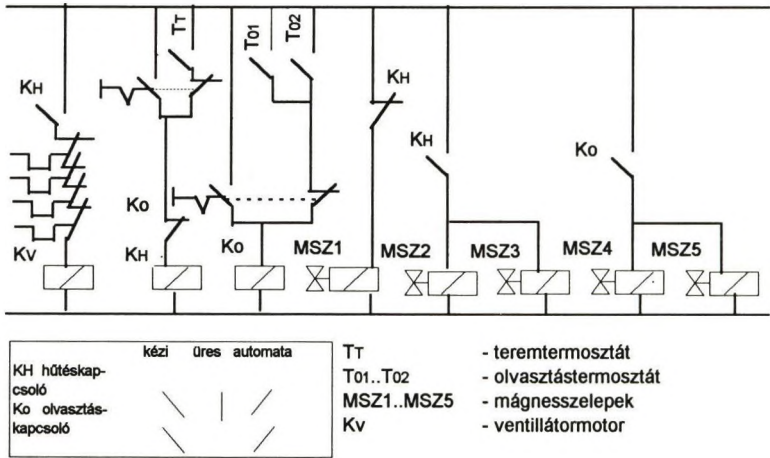
———— ammónia gáz

- - - - - ammónia gáz, folyadék

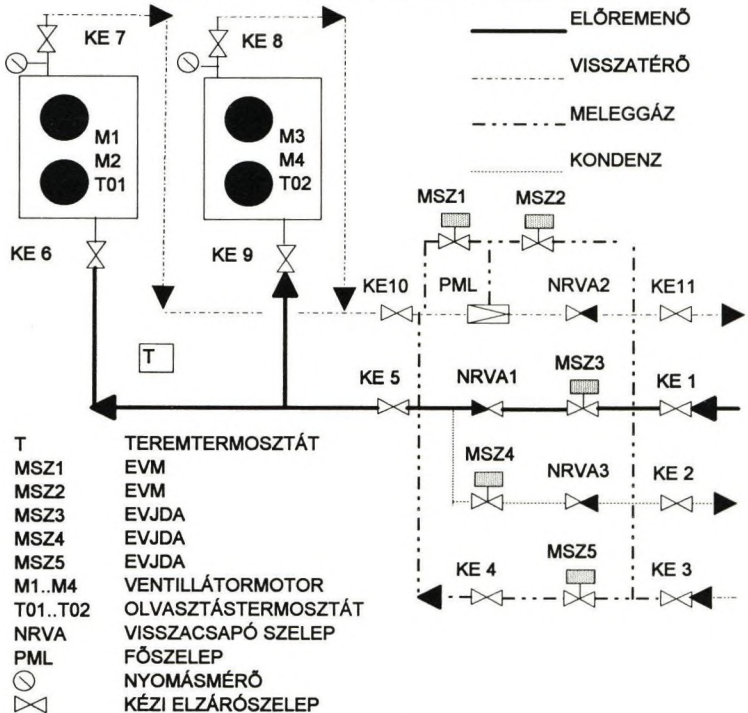


- 1- hűtött helyiség
- 2- kondenzátor
- 3 - folyadékgyűjtő edény
- 4 - folyadékleválasztó edény
- 5 - kompresszor
- 6 - ammónia szivattyú
- 7 - elpárolgató

A HŰTŐTEREM ELEKTROMOS KAPCSOLÁSI RAJZA

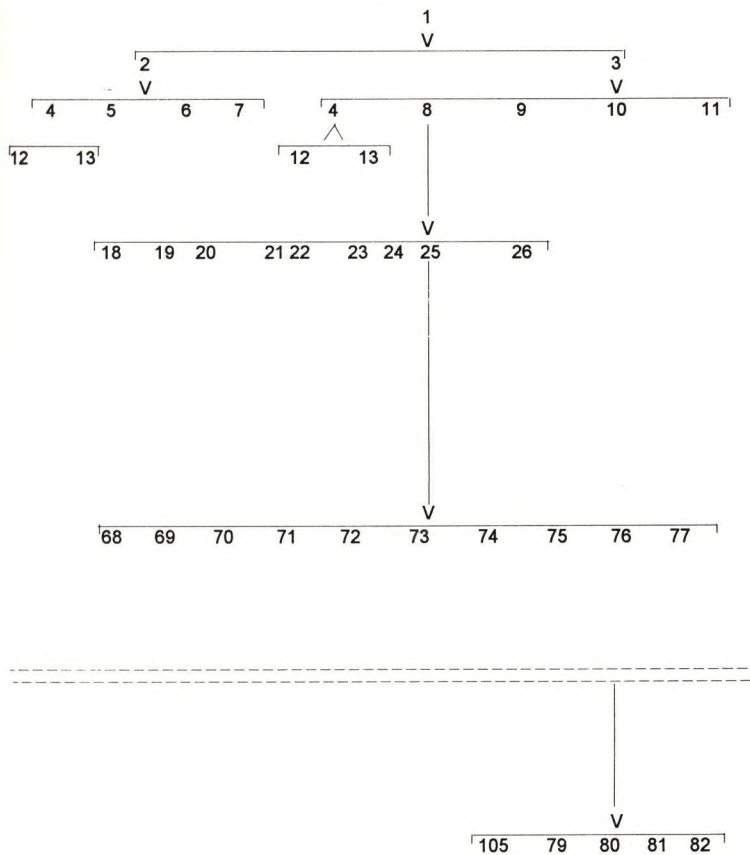


A HŰTŐTEREM KALÓRIKUS KAPCSOLÁSI RAJZA



3. sz. melléklet

HIBAFA /RÉSZLET/



A Logo nyelv oktatása a műszaki pedagógusképzésben
a Széchenyi István Főiskolán
dr. Nyéki Lajos
Széchenyi István Főiskola

Bevezetés

A Logo nyelv oktatásának a műszaki pedagógusképzésben is nagy jelentőséget tulajdonítunk azért, mert fontos eszköznek tekintjük mind az általános-, mind a középiskolai tanulók problémamegoldó gondolkodásának fejlesztésében.

Hallgatóinknak 1985 óta van módjuk a műszaki tárgyak oktatása mellett a technika tantárgy tanítására szakosodni. A nálunk végzett tanárok így egyaránt kerülhetnek általános iskolába vagy középfokú oktatási intézménybe tanítani.

A Logo nyelv oktatásának tantervi keretei

1984 óta oktatjuk a Logo nyelv elemeit "A számítógépek pedagógiai alkalmazása" című tantárgy keretében, a Didaktika tárgy harmadik féléveként, a harmadik évfolyam második szemeszterében.

"A számítógépek pedagógiai alkalmazása" című tárgy féléves programja a következő témakörökből épül fel:

- Terminológiai kérdések (CAI, CMI, CBT, stb.),
- A számítógéppel segített oktatás (CAI),
- Intelligens oktatórendszerek (ICAI, ITS),
- Hypertext,
- Szerzői nyelvek és rendszerek,
- A Logo programozási nyelv elemei,
- A számítógép alapú oktatásirányítás (CMI),
- Feladatlapok és feladatok elemzése,
- Számítógépek a tananyag elrendezésében.

A tantárgyat számítógép kabinetben, heti két órás időkiméretben tanítjuk. A félév gyakorlati jeggyel és vizsgával zárul. A hallgatók a félév során egy-egy számítógépes oktatóprogramot készítenek.

Az 1990. évi tanterv a négyéves képzés utolsó három szemeszterében kötelezően választott pedagógiai tantárgy felvételét írja elő. A felvehető tárgyak egyike "A számítógépek pedagógiai alkalmazása II." címet viseli.

Mivel épít "A számítógépek pedagógiai alkalmazása" nevű tantárgy anyagára, ezért a negyedik évfolyam első szemeszteré-

ben van mód a felvételére.

"A számítógépek pedagógiai alkalmazása II." fakultatív tantárgy tartalma a következő:

- A számítógépes oktatóprogramok készítésének módszertani kérdései,
- Számítógépes oktatóprogramok készítése szerzői nyelv és szerzői rendszer felhasználásával,
- A Logo nyelv lehetőségei a problémamegoldó gondolkodás fejlesztésében.

A tantárgyat számítógép kabinetben, heti két órás időkiméletben oktatjuk. A félév beszámolóval zárul.

A Logo nyelvnek az oktatás során kiemelt jellemzői

Hallgatóink a két féléves "Számítástechnika" tantárgy második félévében BASIC nyelven (korábban GW-BASIC, majd később QBASIC nyelvjárásban) tanultak programozni. Ennek következtében volt összehasonlítási alapjuk a Logo és a BASIC lehetőségeit illetően.

"A számítógépek pedagógiai alkalmazása" című tárgy szükös i-

dókeretében a Logo nyelv következő jellemzőit szoktuk kiemelni:

- a nyelv bővíthetőségét (eljárás-definíciók),
- a rekurzió alkalmazhatóságát, és
- a teknőc-grafikát.

Mindhárom jellemző ütköztethető a BASIC nyelv ismert korlátaival. A Logo nyelv sajátosságainak bemutatására a teknőc-geometria körébe eső egyszerű programokat szoktunk felhasználni (ház, spirál, virágok, stb. rajzolása). A hallgatókat érdekelni szokták a Logo-ban rejlő, számukra addig ismeretlen lehetőségek.

"A számítógépek pedagógiai alkalmazása II." nevű tárgyban mód van arra, hogy a hallgatókkal is készíttessünk Logo programokat.

Az oktatás tárgyi feltételeinek alakulása

Az oktatás 1984-ben néhány ABC 80 és HT 1080Z mikroszámítógéppel indult. 1985-től helyi hálózatba kötött monokrom monitoros HT 1080Z gépekkel felszerelt kabinetben tanítottunk. Az 1988/89-es tanévben már monokrom monitoros C 64-es gépekkel volt ellátva a kabinet. Ez volt az az év, amikor először

tudtunk működő Logo programot bemutatni a Terrapin C 64 Logo interpreter segítségével.

1990-től tértünk át az IBM PC kompatibilis személyi számítógépek használatára (először 8088-as XT-k, majd később 286-os AT-k használatára. Mivel a kabinetekben általában Hercules csatolókártvás monokróm monitoros számítógépek vannak, ezért a CGA-kártyára készült LCSI Logo interpreterünk használata mindig gondokat okozott. A szoftver csak emulátorral működött, a szöveges ablak betelésekor mindig megtelt különböző, nem oda való karakterekkel. Érdekes módon ezek a nehézségek később emulátor nélkül, EGA vagy VGA színes monitorok használata során is jelentkeztek.

Az oktatásban részt vett hallgatók létszámának alakulása

1984-től évfolyamonként egy nappali és egy-két levelező tagozatos tanulókör tanulta "A számítógépek pedagógiai alkalmazása" című tantárgyat. 1994 tavaszával bezárólag 167 nappali és 191 levelező tagozatos hallgatót volt lehetőségünk megismertetni a Logo nyelv elemeivel.

A számítógép alkalmazásának lehetőségei a beszéd-nyelvtanulási zavarok diagnosztikájára épített terápiában a hallássérült előkészítő tanulóinknál az 1992/93-as és 1993/94-es tanévben szerzett tapasztalatok alapján.

**Összeállította: Baharevnyé Nádudvari Katalin
Cházár András Óvoda Általános Iskola Diákotthon és Speciális Szakiskola
Vác Március 15 tér 6**

Az 1992/93-as tanévre beiskolázott előkészítőseink a több éves intenzív óvodai fejlesztés ellenére sem érték el azt a szintet, mely lehetővé tette volna eredményes oktatásukat a hagyományos oktatási módszereket alkalmazva. Értelmileg épek / IQ 80-104 közötti / 6 életévüket betöltötték, de komoly részképesség kiesések miatt speciális fejlesztésre szorultak / analízis-, szintetizálási-, megfigyelőképesség, emlékezet fejlesztés./ Ez viszont rendkívüli fáradékonyáguk, s nem megfelelő fizikai állóképességük, kommunikációs nehézségek miatt nehezen volt motiválható, játékosá tehető.

Keresve új és újabb lehetőségeket arra, hogy ha pillanatnyilag a hangos beszéd elsajátításáról le is kell mondanunk, legalább az írásos beszédet kialakíthassuk a gépirás felé fordultunk.

Mechanikus írógéppel indult a foglalkozás, de azonnal fény derült a gép hátrányaira: a leütéshez nem rendelkeztek kellő erővel, ujjuk könnyen becsúszhatott, beszorulhatott a billentyűzet közé.

A munkát VC 20-as számítógéppel folytattuk. Itt a leütéshez elég az érintés is, s a gyerekek akarata szerint fehéren, vagy feketén megjelenő betűk és ábrák jó motivációt adtak.

Pozitív fejlesztő hatása volt annak is, hogy az esetleges hibákat önállóan tudták eltüntetni. Megszabadultak a kudarc élménytől, melyet sokkal nehezebben tudtak elviselni mint más gyerekek, hiszen tanulási nehézségeik miatt naponta többször átérték eddigi iskolai pályafutásuk alatt.

A gépirás önmagában motivált, hiszen a mozgásnak azonnal jó eredménye volt a betű szép képében. A sikerélménynél jobb motiváció nincs, e mellett pozitívan hatott a gyerekek önértékelésére is.

A gépirás az írásirányok betartásában, a kiejtéstaniásban, az analízis és szintetizálási képesség gyakoroltatásában, az esztétikai érzék, ritmusérzék, szukcesszív emlékezet fejlesztésében komoly szerepet játszik. E fejlesztő hatásokat vettük figyelembe mint elsődleges célt, s a gépirás megtanulását mint ajándékot értékeltünk.

A gép megkönnyítette a kis és nagy betűk egyeztetését is. A tetszetős külalakú írást a gyerekek hazavihették, s megörvendeztethették vele szüleiket előkészítő gyerekekről lévén szó /6-7 évesek / a szülők még nem tudták igazán feldolgozni azt a traumát, melyet a sérülés hozzáfűzéseként átéltek, így a gyermek minden pozitív eredménye hozzásegített a szülő-gyermek kapcsolat jobb tételéhez.

Első feladatunk volt a számítógéppel való ismerkedés. A kurzor mozgatásával az irányok gyakorlásának új lehetősége adódott. Perceken belül ellesték a különböző billentyűk kezelését és a legkülönbözőbb sormintákat állítottak elő.

Megtanultuk a gépírásnál alkalmazott tornákat. Táblákat készítettünk az egyes ujjak leütési területeiről, egyeztettük a kezeket és az ujjakat a táblák színei segítségével.

A gépelést a helyes ujjtartással tanultuk, s arra törekedtünk, hogy a tanult szavakat azonnal le is tudják írni a gyerekek, s nem fordítottunk kellő figyelmet a technikai rész gyakoroltatására, ahogy ezt a gépírást tanító módszertani könyvek javasolták.

Ma már tudjuk a gyakorlatból, hogy a technikai rész gyakoroltatására fordított idő busásan megtérül, és nem kell annyira törekedni értelmes szavak gépeltetésére. /2. számú gyakorlólap /

Első értelmes szavunk a mama volt. Ezt két kézzel kell gépelni a gépírási szabályok szerint, s a gépírástanítás módszertana más szórendet javasolt, de a gyerekeket a szó erős érzelmi töltése átsegítette az akadályon.

A tea, te fa baba szavaink melyek a mamát követték már jobban alkalmazkodtak a gépírási módszertani követelményeihez. Szavaink válogatásánál alapvető követelmény volt, hogy ismerjék jól a szót, s ha kiejteni nem is mindenki tudja, de ujjábécével jól begyakorolt szó legyen. /Sajnos gépünkön ékezetes betűk nincsenek ezért néhány szót mellőzni kényszerültünk. /

A szavakat képről és kézzel aláírt szóképről gépelték, míg a gép nyomtatott nagybetűket „a monitor kisbetűket ad. Azt meglepően hamar elsajátították, hogyan lehet gombnyomással a betűtípust megváltoztatni, s az egyeztetés így egyéni fejlődésnek megfelelően, észrevétlenül, de eredményesen megtörtént. /1. típusú feladatlap /

Az értelmes szavak gépeltetésének sittetésével úgy véltük az írásos beszédnevelést tudjuk hatékonyan támogatni, azzal is, hogy a következő lépésként képről gépelték a szavakat, de a lap hátoldalán ott volt az írott szókép is, de kevert sorrendben. Egy szóból 3-3 sort kellett gépelniük. /3-4. típusú feladatlap /

Örömmel versenyeztek ki tudja leírni a szót anélkül, hogy a lapot megfordítaná? Külön lelkesedéssel kerestek olyan szavakat melyek tartamilag kapcsolódtak a lap szavaihoz, de nem tudjuk leírni őket az ékezetes betűk hiánya miatt.

Néhány szó az első év eredményeiről: legfontosabb, hogy megszerették a számítógépet, s a velük kapcsolatban lévő felnőtteket saját példájukkal győzték meg arról, hogy a számítógép kezelése nem ördögösség.

Hetente kb 1 óra gyakorló idő jutott egy tanulóra a napi 10 perces gyakorló időből összeadódva. Ezt az időt a nevelőtanár F. né Bihari Gabriella segítségével tudtuk biztosítani a korrepetálási órák megosztásával / két helyszínen folyt a munka korrepetálás, gépelés egyidőben/. Eredményeinket látva lehetővé vált, hogy a számítógépes teremben együttesen is gyakoroljunk a második év februárjától. Ott több

Commodore 64-es gép van, szövegszerkesztő programunk is van hozzá, így ki is tudjuk nyomtatni a műveket, s a gyerekek is már tudtak önállóan állandó felnőtt kontrol nélkül dolgozni. Gyakorló időnk így ugrásszerűen megnőtt, a mozgások automatizálódása is hamarabb várható.

Március közepén iskolánk egy darab IBM AT 286-os számítógéphez jutott, s lehetőséget kaptunk, hogy a Mesevilág programmal dolgozhassunk heti 1-2 órában. Osztálylétszámunk kicsi így egyszerre mindnyájan odafértünk a géphez. A gyerekek bámulatos gyorsan ellesték a billentyűzet kezelését, s ma már képesek differenciáltan is dolgozni a programmal. Erre a program lehetőséget ad: nyíllal és a space billentyűvel, a példát látva, vagy teljesen önállóan gépelik be a kívánt szót. Örömmel töltött el bennünket az, hogy a gyerekek önként választják a nehezebb megoldási módokat, és a gépelésnél ragaszkodtak a helyes ujjtartáshoz.

A Mesevilág program háttérbe szorította a gépírás mechanikus gyakorlását, de mivel célul a beszédfejlesztést tűztük ki nem bántuk.

Nagy lelkesedéssel kezdtük a Betűvarázs programmal való ismerkedést is májusban. E program gazdag szókészlete komoly gondot okozott, de a gyerekek szeretik az állatokat, s nem bánták a nehézségeket.

Integrációs zavarral küzdő tanulóinknál a tárgykép, szókép kapcsolat kiépítését a számítógép nagyban segíti. Mindkét program változatos sorrendben gyakoroltat, a hibák gombnyomásra eltűnnek, a jó eredményt azonnal értékeli a gép. Türelmes, színes, sokkal érdekesebb gyakorolni vele mint a papírképekkel

Kaptunk két óvodások számára készült forma, szín és irányokat gyakoroltató programot is a kollegáktól a tavaszi továbbképzésen. A gyerekek nagyon szeretik, és fejlesztő hatásuk jól gyümölcsözne ha több lehetőségünk lenne a gyakorlásra.

Májusban kaptuk meg Szegedről a speciális siketek számára készült gépírást tanító programot. / IBM gépre kisiskolásoknak és nem előkészítősöknek./ A program a felnőttektől komolyabb számítógép kezelési ismeret vár a felnőttektől, ezért betanulásunk még most is tart. A program nyitott a tanár maga állítja össze mely szavakat akar gyakoroltatni, így a napi munkát jobban tudja segíteni. Mivel fejlesztő hatásait; /szukcesszív emlékezet, önállóság, írásosbeszéd / nem nélkülözhetjük a nehézségek ellenére használjuk.

Úttörő munkánk eredményeként a jelenlegi előkészítő osztályunk már a mi tapasztalatunk alapján összeállított tanmenet szerint folyik a fejlesztés, a gépírás a mi VC 20-as gépünkkel. Kár, hogy nincs a géphez kész játékprogramunk.

[MG1]

Törtely Éva:

Az informatika játékos tanítása Pilisvörösváron

Hazánkban szinte mindegyik iskolában ma már számítástechnikát/informatikát is tanítunk. Szerencsére a jobb minőségű számítógépek terjedésével egyre inkább az oktatás tartalmassága és színvonala is jobbá válik. Abban a szerencsés helyzetben vagyok, hogy egy olyan iskolában tanítok, ahol ebben a tanévben bevezethettem az informatika oktatását. A múlt tanév végén a Pilisvörösvári Templom téri Általános Iskola igazgatójának (Gordos Dénes) felkérésére, és vele együtt dolgozva hozzákezdünk az iskola informatika tagozatának megszervezéséhez.

Első feladatunk a tanítási koncepció megválasztása volt. A játékos informatika tananyag- és módszer-rendszer az általános iskolában kiválóan alkalmas. Ezt jól ismerem, hiszen kialakításában a kezdetektől részt veszek. Hét év óta tanítom a kisgyermekeket informatikára is.

Második feladat a szakterem kialakítása volt. Nem mindegy, hogy milyen helyiségben, milyen gépek elé, milyen bútorok közé ültetjük gyermekeinket.

Nagy gonddal válogattuk ki a bútorokat. Az én termemben a székek többféle helyzetbe állíthatóak attól függően, hogy mekkora gyerek ül a munkaasztalhoz és attól függően, hogy mi az óra feladata. Ha az asztallapon írunk, tervezünk (füzetbe, ceruzával), akkor a kisgyerekeknek teljes magasságra állítjuk a székeket. Az egy szinttel lejjebb levő billentyűzet kezeléséhez alacsonyabb magasság is elég.

A háttámla távolsága és dőlésszöge is állítható. Ez azért is fontos, mert a gép előtt ülve egyre több időt töltenek a gyerekek, s a nem megfelelő testtartás egész

életre szóló torzulásokat okozhatna.

Az asztalok is saját elképzelés szerint készültek. Az asztal felülete olyan nagy, hogy a számítógépek és a monitorok (amelyek nem a gép tetején állnak) mellett a könyveknek, füzeteknek is jut hely. Lábtartóval láttuk el a padokat, hogy ha a székről nem ér le a gyermek lába, akkor is tudjon támaszkodni. A lábtartó különböző magasságokban helyezhető el a gyermek méretéhez igazítva.

Három kihúzható polca van minden asztalnak: a középsőn a tasztatúra helyezkedik el, a két szélső pedig az egér számára készült. (Mindkét oldalon azért szükséges "egérpolc", mert minden osztályban lehet balkezes gyerek.)

A legtöbb szakteremben az asztalok körben a fal mellett találhatóak. Amikor az elképzeléseinket meg akartuk valósítani, akkor jöttünk rá ennek egyik lehetséges okára. Ugyanis villanszerelési szempontból ezt a legkönnyebb megvalósítani. Komoly nehézségekbe ütközött annak megoldása, hogy a két sorban lévő asztalokhoz hogyan vezessék el az áramot. A védett tokozott vezetés a padlón és az asztalokon valamivel több költség, mint a falon vezetve, de a mi döntő szempontunk a tanítás hatékonysága. A gépteremben is jobbnak tartjuk, ha a gyerekek egyirányban ülnek. Ennek előnyei: frontális munka is megvalósítható, nem fordítanak a gyerekek hátat a tanárnak és egymásnak, a természetes megvilágítás mindenki számára optimális irányból érkezik.

A szakteremben tíz tanulói munkahelyet kívánunk kialakítani, de jelenleg csak nyolcra "tellett". Nyolc 386-os számítógépünk van és egy lézernyomatónk. Az óra vezetéséhez elég ez a nyolc számítógép, viszont a tanulók egyéni munkájához jó volna több gépet beállítani.

Fontos, hogy jó minőségű monitor elé ültessük a gyerekeket. Ma már nem

álom az ilyesmi. De a monitorokat lássuk el szűrővel is. A szemünk fénye a gyerek, vigyázzunk hát az ő szemére is.

Külön tanári munkahelyet is kialakítunk majd, ahol a legjobb minőségű legtöbbet "tudó" számítógép áll. Korszerű informatikai műveltséget csak korszerű eszközökkel lehet közvetíteni, átadni. Erre a célra egy PENTIUM gép beállítását tervezzük. (Majdnem minden iskolában ott állnak és porosodnak a régi - Primo, HT és egyéb - "öskövületek". Mi lesz a régi gépek sorsa?)

Persze, nemcsak számítógépekből áll a szakterem berendezése. A hagyományos eszközöket is használjuk. De a tábla is korszerű: filccel lehet rá írni. Jó, ha nagy papírlapokat is fel lehet rá függeszteni - így előre lehet készíteni feladatokat, rajzokat.

A falra kerülnek kifüggesztve a hasznos tudnivalók, "puskák". Nem az a fontos, hogy a gyerek fejből tudja az utasításokat, hanem az, tudja hol keresse, s tudja alkalmazni azokat. A falon jut hely a gyermekek munkáinak kiállítására is.

Az iskolának régebb óta van videokamerája. Miután a video az egyik legnépszerűbb információtechnikai eszköz napjainkban, ezért természetesnek érzem, hogy a videózással is foglalkozunk. Nem csak jól megválasztott -például az informatika legújabb eredményeiről szóló - felvételek megtekintésére gondolok, de felvételeket is készítünk, amelyeknek szereplői, írói, rendezői és operatőrei a gyerekek.

Szintén a teremben helyeztem el a hangszer minőségű szintetizátort is, amelyet örömmel használunk.

A terem lényeges - a jövő informatikájának bemutatása, tanítása szempontjából nélkülözhetetlen - berendezése a telefon és a számítógéphez kötött telefonvonal is. Fontosnak tartjuk, hogy a gyerekek az informatikának a

számítógéptechnika, a robotika és a többi eddig említett területe mellett a telematikával is foglalkozzanak. Az elektronikus levelezés ma már gyakorlat. Ezt ugyanúgy célszerű egy kisgyermeknek megismerni, mint például az üzenetrögzítő vagy a rádiótelefont. Pilisvörösváron kiemeli az elektronikus levelezés oktatásban való alkalmazásának jelentőségét az is, hogy a gyerekek így a német nyelv gyakorlására egy vonzó élő szituációt próbálhatnak ki, német iskolásokkal folytathatnak számítógépes kommunikálást.

A pilisvörösvári iskolában egyelőre csak két első és egy második osztály tanul informatikát. A gyerekeknek heti három órájuk van. Az osztályokat két csoportra osztottuk. Amíg az egyik csoport informatika órán van, addig a másik csoport német órán vesz részt. (Ez a nyelvtanítás szempontjából is ideális.) Így egyszerre 10-12 tanulóval foglalkozom. Az elsősöknél a csoportbontás teljesen a véletlen műve, névsor szerint bontottunk. A másodikosoknál a németet tanító kolléganő döntött a tavalyi tanév tapasztalatai alapján.

A JIO (Játékos Informatika Oktatás) koncepciót már több helyen bemutattuk. Több könyv és cikk foglalkozik vele. Ezek közül talán a Játszd el a teknőcöt! sorozatra hívnám fel a figyelmet, melynek három kötete részletesen leírja a JIO tananyagát egytől ötödik osztályig. (A többi korosztály részére alkalmas tananyagról a publikáció készül.)

Legfontosabb eleme a JIO-nak a Logo-pedagógia. Ezzel az elnevezéssel S. Papert könyvében találkozhatunk. Ezt a könyvet kötelező olvasmánynak írnám elő minden pedagógus számára.

Papert meglehetősen éles kritikát mond az iskolákról. Ezen érdemes elgondolkodni. Véleményem szerint a magyar iskolákra is vonatkozhat ez a kritika.

A Logo-pedagógia / JIO egyik célja, hogy a gyermekeket megóvjuk a különféle fóbiák kialakulásától.

Számomra a Logo-pedagógia nemcsak a Logo tanítását, hanem egy új szemléletmódot, mindenekelőtt új pedagógiai attitűdöt jelent. A pedagógus legyen a gyerek partnere a tanítási-tanulási folyamatban. Minden gyermek óriási tanulási vágygal érkezik az iskolába. Sajnos ez a tanulási vágy a kudarcok miatt hamar eltűnik. A hibák "kezelése" nagyon fontos ebben a folyamatban. Egy-két rossz nevelői reakció után a gyerek szorongani kezd, nincs sikerélménye. Természetesen a hibákat javítani kell, de nem mindegy, hogyan. Ez a nevelő igen fontos feladata. "Nem baj, hogy hibáztál, de a hibát javítsuk ki!" Ez egy lényeges mondat. A hibát meghagyni, az számít hibának. Ha ezt az attitűdöt sikerül más órákon is alkalmazni, akkor az oktatás-nevelés minőségileg változik. Ezért tartom szerencsésnek, ha egy tanító megismerkedik ezzel a módszerrel, s az osztályának informatikát is tanít. Így a tanítás egészében tudja érvényesíteni a Logo-pedagógiát. (A JIO kialakítása során több budapesti iskolában végeztettünk szorongás-tesztet, s minden vizsgált iskolában az informatikát tanuló osztályokban a szorongás kisebb mértékű volt, mint a kontroll osztályokban.)

Sokféle törekvés van az iskola "humanizálására". (Furcsa: az iskola nem humánus intézmény?) Véleményem szerint a JIO között figyelemre méltóan hatásos, sikeres. Ezt a törekvést eddigi iskolakísérleteinkben nem fogalmazzuk meg ilyen markánsan, de a célok között ez mindig a kiemelték között volt.

Már a szakterem kialakításánál írtam a "puskázás" lehetőségéről. Nem biztos, hogy az a jó, ha ismeretekkel telezsúfoljuk az elménket, de egy probléma megoldásánál csődöt mondunk. (Lenyűgöznek a tévében látható "eleven lexikonok", de nem állítom, hogy mindenkinek ilyennek kell lenni.)

Az iskolába bekerülő gyermek még tulajdonképpen nem "igazi" iskolás. Az óvodai játékos foglalkozások után még nehezen illeszkedik az iskola rendjébe. Ezt sajnos sok kolléga nem veszi figyelembe. Az első időben az ismereteket játékosabb formában kell neki átadni. Ez a JIO lényege is.

Sokan elhűlnek, mikor megtudják, hogy a hatéves kisgyerekek informatikát tanulnak. De miután ez a gyermeknek játék, nem megterhelő. (Fogadóórán megkeresett az egyik szülő, hogy mikor fognak már végre tanulni, mert a kislfia mindig azt meséli otthon, hogy játszottunk az órán. Az iskolai tanulásról alkotott elképzelésébe nem illett bele a játékos tanulási tevékenység.)

A pilisvörösvári osztályokkal kezdetben nagyon sok mozgásos játékot játszottunk, szinte az egész óra ebből állt. Az egyik játékunk a bal-jobb játék volt. Azt a kezét emeli a magasba, amelyiket mondom. Megtanultuk a balra át-ot, jobbra át-ot. Megállapodtunk abban, hogy mekkora egy robotlépés. Ezek után megszemélyesítettük a robotokat. Mivel szeptemberben még szép volt az idő, ezeket a játékokat az udvaron játszottuk.

Nagyon fontos a kisgyermek számára az irányok biztos ismerete. Még a másodikosok között is sokan vannak, akik tévesztenek.

Később nagy nyilakkal (kartonból készítettem) kiraktuk a robot útvonalát. Ezt az utat kellett a robotnak végigjárni. Ez már nehezebb játék, hiszen az irányítók nem mozognak. Ez a fordulásoknál jelentett problémát. "Képzeld magad a robot helyébe!" A kicsiket arra buzdítottam, hogy álljanak ők is ugyanabba az irányba, mint a robotot megszemélyesítő társuk. Így már könnyebb volt számukra a helyes irány meghatározása.

Többféle programozható robotot sikerült beszereznünk. Ezek minősége és tudása igen széles skálán mozog. (Van orosz gyártmányú Elektronikánk, angol

ROAMERünk és - kölcsön - gömbrobotunk.) A gyerekek nagy örömmel használják ezeket. Azért is jó, hogy ilyen sokféle eszközzel rendelkezünk, mert a tanítványaink felfedezhetik, hogy mi bennük a közös, s észreveszik a különbségeket is. Számomra is érdekes volt, hogy egy eszköz bemutatása után bátran játszottak a másikkal is, egyből felismerték a hasonlóságot. A transzfer hatás itt is érvényesült.

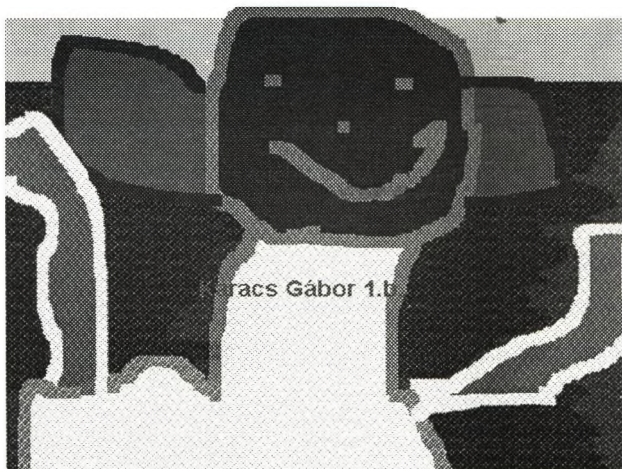
Ezekkel a játékokkal készítjük elő a képernyőteknőccel való munkát. Erre körülbelül egy hónap után került sor. Természetesen a kicsikkel a Kislogót használjuk, amely az ő számfogalmuknak megfelelő.

A második osztályosok is idén kezdtek informatikát tanulni, ezért velük is ilyen sorrendben haladtam. Mivel azonban ők már egy évet jártak iskolába, gyorsabb ütemben lehet velük haladni. Olyan robotutakat építünk, amelyek zárt alakzatokat alkotnak. Ismerik a négyzet, téglalap, háromszög fogalmát. Mivel a háromszög lejárásához, rajzolásához nem elég a jobbra át, szükség van az elfordulás mérésére is. Ehhez is készítettem segédeszközt: egy körlapot tizenkét egyenlő cikkre osztottam. Így a padlóra a körcikkek és a nyílak segítségével bonyolultabb alakzatokat is ki tudunk rakni.

Az elsősök a képernyőteknőcnek csak egy-egy betű és szám beírásával parancsolnak, a másodikosok viszont teljes szavakat használnak. (m 4 = megy 4, ami a forward 40-nek felel meg.)

Itt mutatkozik meg a számítógép másik jótékony hatása: pontosságra szoktat. Csak a helyesen leírt parancsszavakat érti meg a gép. (Az nem igazán lepett meg, hogy a jobbra szót sokaknak nem sikerült elsőre hibátlanul leírniuk, de több gyerek hosszasan próbálkozott azzal is, amíg - bara, barra után - sikerült helyesen leírni a balra szót.)

Természetesen nem csak a Logo tölti ki az órákat. Sok más programot is használunk. Nagyon kedvelik a gyerekek a Math Rabbit, Reader Rabbit játékos oktatóprogramokat. Nagy élvezettel rajzolnak a Paintbrush-sal. A másodikosok már önállóan el is tudják indítani ezeket. Számomra is nagy élmény volt figyelni őket, ahogy a kezdeti firkák után egyre inkább felismerhető rajzok születtek. Már elég nagy biztonsággal tudják kezelni a programot. Az egyik legnagyobb öröm a számukra, ha a jól sikerült képet kinyomtatva hazavihetik. Sajnos az iskola nem rendelkezik színes nyomtatóval - így a fekete-fehér kép nem mindig adja vissza az eredeti szépségét.



gyermekrajz

Nagyon fontos szerintem, hogy az órákon többféle dologgal is foglalkozunk. Minden óra elején van mozgásos játék, játék különféle eszközökkel, beszélgetés és csak utána ülünk a számítógépekhez. Van, amikor kötött az óra, kapnak egy

feladatot, azon dolgozhatnak óra végéig. Néha kevésnek bizonyul a negyvenöt perc, ilyenkor a féligkész munkát rögzítjük és valamelyik következő órán folytatjuk azt. Van olyan óra is, amikor szabadon dönthetnek, ki mit szeretne csinálni, mivel szeretne foglalkozni. Van olyan csoport, ahol a gyerekek inkább rajzolni szeretnek, van olyan, ahol a "nyuszikással" akarnak inkább játszani, és nagy örömmre van olyan csoport is, ahol a többség a képernyő teknőcöt irányítgatja.

Egy géphez két gyerek ül. Ezek nem állandó párok, óra elején választják ki, ki kivel szeretne dolgozni, de megfigyelésem szerint - főleg a másodikosoknál - kezdenek állandó párok kialakulni.

Az informatika nem tartozik a hagyományos tantárgyak körébe, éppen ezért nehéz a hagyományos módon értékelni is. Az én véleményem szerint jó lenne, ha mindenütt sikerülne az osztályzást elhagyni, s a tanulókat szöveggel értékelni. Erre főként az alsó tagozaton van szükség.



Farkas Károly: Logo nyelv és logo-pedagógia

Seymour Papertről az egyik informatika konferencián azt hallottam, hogy ő a ma élő legnagyobb pedagógus. Aki olvasta a Mindstorm, magyar fordítás címe Észrengés (1) könyvét jól tudja, hogy ő valóban többet tett, mint egy, a gyermekek számára is használható programnyelv megalkotása, ő valóban új pedagógiát alkotott és hirdet. A következőkben megfogalmazott téziseimet, hipotéziseimet zömében Papert nyomán alakítottuk ki. Szinte mindenben követjük őt, néhány részletben pedig megpróbálunk még jobban elrugaszkodni, még inkább a jövőbe tekinteni, a jövő évezredre felkészíteni. (Bár egyesek szemében már Papert is néha utópisztikus.)

A logo-pedagógia tehát Papert nevelésről, oktatásról vallott nézetei. Ezek természetesen a Logo programnyelv alkalmazása köré épültek ki.

A Logo nyelvről megjelent legátfogóbb könyvet hazánkban Votisky Zsuzsa szerkesztette (2).

A Logo nyelvet én a kisgyermekoktatására igen értékesnek találtam. A Logo az egyes szakterületeken természetesen nem a leghatékonyabb programnyelv, ez inkább csak egy játék, egy univerzális, a játékokkal szembeni pszichológiai igényeket mintaszerűen kielégítő játék. Egy olyan eszköz, amely a gyermek számára a játszást és a tanulást szinte megkülönböztethetlenné teszi.

A Logo főszereplője a teknőc. A teknőc egy tollat cipel magával, ezzel vonalat húz maga után, haladási útvonala látható lesz. Ez a képzeletbeli és kibernetikus eszközként avagy LEGO elemekből megépített játékként is létező eszköz, "lény" nemcsak a kisgyermekoktatás, hanem a programozással ismerkedő felnőttek számára is kiváló gondolkodási fogódzó. A teknőc helyére képzelve magunkat a szintónia, a beleélés, segít a mozgáselemek megfelelő sorba rendezésében. A teknőc-geometria - amelyet a Logo programnyelv fejlesztett ki - korántsem csak a geometriai ismereteket fejleszti, hanem igen hatékonyan például az algoritmikus gondolkodási módot is.

A Logo nyelvre jellemző további tulajdonságokat néhány blikkfangos szóval is sorolhatjuk, ezek: a "Rubik-hatás", a "LEGO-hatás", a "keresztbe fektetett dominó elv".

A "Rubik-hatás"

"Lehet, hogy egy feladat, amelyen gondolkozol, egyszerű, de ha felkelti az érdeklődésedet, mozgósítja találegonyságodat, és végül, ha sikerül önállóan megoldanod, átéled a felfedezés izgalmát és diadalát" írja Pólya György a "Gondolkodás iskolája" című könyv előszavában. A teknőccel való rajzoltatás a Rubik-kockához hasonlóan vonzó, könnyen átlátható, egyszerűnek tűnő, de végtelen sok variációt lehetővé tevő, fejlett térszemléletet, kreativitást, igen fegyelmezett, kitartó végrehajtást kívánó kihívás. Az alkotás élménye a felfedezés izgalma és diadala már a programnyelv elsajátításának kezdetén is élvezhető. Már a Logo két alapparancsával is - menj előre valamennyit, fordulj jobbra valamennyit - a felhasználó önálló, alkotó munkába kezdhet. (Például: menj 50 jobbra 120 háromszor ismételve eredménye egy háromszög.) Megfelelő algoritmus alkalmazásával bármely síkbeli alakzat megrajzolható. A munkánk eredménye vagy sikertelensége azonnal látható. A Logo interaktivitása kiemelkedő, így próbálgatásaink bőséges lehetőséget adnak a heurisztikus tanuláshoz. Ezért természetes, hogy a gyermekek többsége, sőt a számítástechnikától ódzkodó felnőttek közül is sokan, sőt a számítástechnikától elidegenített diákok is a teknőc-geometria elemeinek birtokában gyakran számukra élvezetes önálló alkotásba, tanulásba kezdenek.

"LEGO-hatás"

Az áttekinthetőség, a gondolkodás, az alkotás eredményének vizuális megjelenése mellett a Logo azért is hasonló a LEGO játékhoz, mert mindkettő korszerű technikatudományi elveket hordoz. Mindkettő a rendszerszemlélet, a strukturált gondolkodás, a modul vagy szekrényelv használatát kívánja és fejleszti. Az építmények összerakása vagy a teknőcgrafikával készült rajzelemek felhasználásával történő konstruálás - próbálgatással,

hipotézisek felállításával, kipróbálás után alkalmazásával vagy elvetésével, a tapasztalt eredmények feldolgozásával, interiorizációjával, a tanulási tempó egyéni megválasztásával, folytonos visszacsatolásokkal, megerősítésekkel, autonóm vezérléssel - fájdalommentesen történik.

A "keresztbe fektetett dominó" elv

Ha a dominó tégláscskáit egymás mögé megfelelő módon állítjuk, az első felborítása a többi eldőlését is eredményezi. Megszakad a lánc, a folyamat, ha csak egyetlen helyen is hiba van. Ki ne tudná: az egész sort részekre célszerű bontani egy-egy keresztbe rakott dominóval. Szakaszonként érdemes a működést kipróbálni, majd csak ezután felállítani a keresztbe tett dominókat és így összefűzni a rendszert. Hosszú programot hibátlanul megírni nem lehet! Emberek vagyunk és nem gépek. Harminc-negyven sornál hosszabb program áttekintése már nagyon nehéz. "Felbontás és összerakás fontos szellemi műveletek." írja Pólya. A Logo-val néhány soros eljárásokat írunk, (a négy-öt sornál hosszabb eljárás már nem is igen lehet jó), amelyek önmagukban is komplettek, kipróbálhatók, működtethetők. Az így kifejlesztett elemekből építhetjük meg a bonyolultabb rendszert, előre gyártott elemekből építkezhetünk. A gyerekek számára ez különösen fontos.

A Logo programnyelv képes folyamatosan megújulni.

A Logo a többi mesterséges és természetes nyelvhez hasonlóan, az átlagosnál sokkal rugalmasabban és dinamikusabban változó, fejlődő rendszer. A számítógéppel való kommunikáció általánosan elfogadott, terjedő módjait nagyfokú flexibilitása miatt tudja, céljai miatt akarja alkalmazni. A technógeometriát már kezdetben is erős listakezelési lehetőség egészítette ki. A lemezkezelés, a DOS alaputasítások is természetesen részei a Logo-nak. A szövegszerkesztés általános fejlődése a Logo változatokban is tapasztalható (A LogoWriter például, amely hazánkban jelenleg bizonyára a legerterjedtebb Logo változat, szerintünk a

kisgyermek számára a legjobb szövegszerkesztő, a Windows alatt futtatható Logo változatok pedig, mint például a WinLogo vagy a ComLogo teljes értékű szövegszerkesztők is.) A legújabb Logo változatok a Macintosh LogoWriter vagy a ComLogo az égerhasználat előterbe helyezésével is könnyíti a géppel való kommunikálást.

A ComeniusLogo-ról

A különféle Logo változatok között jelenleg a leglátványosabb, a kisgyermekekhez leginkább igazodó, az interaktivitást és a Windows környezetet hatékonyan kihasználó a Comenius Logo. A programot pozsonyi szakemberek (számítógép programozók, egyetemi tanárok, kutatók) Felix Blahó, Ivan Kalaš, Peter Tomcsányi készítették. A magyar adaptáció kialakításában Farkas Károly és Törtely Éva vettek részt. A program működtetéséhez legalább 386-os IBM kompatibilis számítógépet célszerű választani. A központi memória legalább 2 Mb legyen, a Winchesteren 1 Mb szabad hely szükséges.

Informatikai szakembernek nem is kellene felhívniuk a figyelmét arra, hogy a programban benne van minden információ, csak elő kell hívni a "Súgó"-t, csak vizsgálgatni kell a különféle példákat.

Logo vagy Windows?

Az, hogy informatikát tanítsunk, ma már nem vitatható. De, mi legyen az informatika tantárgy ismeretanyaga? Ahány informatikatanár annyiféle a válasz. Szerencsére a sokrétűségéből egyre nagyobb rész kerül az általánosan elfogadott halmazba, szakmai körökben egyre teljesebb a konszenzus. Alig akad, aki ne vallaná: a számítógéptudomány az informatika tantárgy jelentős része. Nekünk (a Játékos Informatika tanítóinak) az általános iskola alsó tagozatán célunk, a számítógépet amikor csak lehet mellőzni, de mivel egyre több területen a gépesített információkezelés - amelynek központi eleme a számítógép - a leghatékonyabb, de mivel egyre több technológiában már csak a számítógépes vezérlés használható, és ugyanakkor

a gépekkel való kommunikálás módja egyre inkább hasonlít az élő nyelvre, így abszolút volumenében mi is egyre többet foglalkozunk a számítógéppel.

Mit tanítsunk a számítógép tudományból, a számítógép használatából?

Sokan mondják: felhasználói programok kezelésében alakítsunk ki készséget. Ezzel mi így értünk egyet: ezt is. Napjaink általános felhasználói környezete a Windows. Persze, hogy például a Windowsban található programozott tananyaggal a Tankönyvvel, nem baj, ha az Aknakereső játékkal, és még inkább a Paintbrush rajzoló programmal is foglalkozunk az informatika órákon. A számítógépes könyvpiac egyik slágere napjainkban az inkább gyermekek számára készült könyv (3), amely főként a Paintbrush használatába vezeti be az olvasót.

Ugyanakkor kezdetektől fogva létezik a számítástechnika, majd informatika tanárok nem kis hányadát tömörítő csoport, amelynek tagjai szerint a számítógép több, mint a tévéjáték, és a minden eddigi eszköznél jelentősebb gondolkodásfejlesztő hatását leginkább a programozás tanulásával, gyakorlásával használhatjuk ki. Természetesen mi sem kívánjuk, a közoktatásban a programozó képzést. Mi is jól látjuk, a jövőben elenyészően kevés programozó szakemberre, ugyanakkor az általános műveltség részeként, mindenki számára, informatikai kulturáltságra van szükség. A programozás fejlesztő hatását kívánjuk kiaknázni. A programozásnak csak alapjait, elemeit tanítjuk, amint az általános iskolában a matematikának is csak parányi hányada a tananyag. Erre a célra napjainkban az egyik legalkalmasabb programnyelv a Logo. A Logo nyelvet alapvetően két irányban fejlesztik: egyre sokoldalúbb, egyre könnyebben kezelhető játék, szinte már felhasználói program és ugyanakkor hatékony programnyelv.

Az informatika oktatás kiváló szakembereinek sikerült a két törekvést (felhasználó szemlélet és gondolkodás fejlesztés) ötvözni. A Comenius Logo mindkét cél megvalósítására javasolható. Olyan kézhez illő játékszer, amelyet teljesen is használhat a gyermek, de ha akarja szét is szedheti, megnézheti a belsejét, és újra konstruálhatja. Egy LEGO elemekből

összerakott látványos játék, amelyet összerakva adunk a gyerek kezébe. Az utóbbi években a Logo szakemberek nemzetközi találkozóin elfogadott minősítés szerint a Comenius Logo a leglátványosabb Logo változat. A már említett sikerkönyvet kiadó Holland cég ezt a Logót "Superlogo" néven terjeszti a világban. (A terjesztési joguk a volt szocialista országok területére nem kizárólagos.) A superlogonak a társalgási nyelvét természetesen a felhasználó választja meg. Így a program használatakor a (piktogramokon túl), az alkalmazható parancsszavak, az üzenetek angol, vagy német, holland, görög stb. nyelven jelennek meg. (Az alapváltozat az angol, angolul mindegyik teknőc ért.) A pozsonyi alkotókkal együtt elkészítettük a magyar változatot is.

A magyar nyelvet használó (célszerűen a magyar Windows alatt futtatható) Comenius Logo egyszerre alkalmas a Windows használatának tanítására, a Logóban készült és készülő oktatóprogramok és játékok alkalmazására, és a Logo programozás tanulására is, így kiválóan a logo-pedagógia számítógépes megvalósítására.

A Comlogo a korábbi Logo változatoknál lényegesen gazdagabb. Körülbelül háromszáz logoprimitív (alaputasítást) tartalmaz. A parancsszavak készlete leginkább a PC Logo nyelvjáráshoz hasonló. A program kihasználja a Windows környezet szolgáltatásait, például úgy mint a Berkley (más néven MSW) Logóban az elkészült rajzok a gördítősávon levő csúszkával mozgathatók. A teknőcök teremtése, mozgatása az egér segítségével is történhet. Négyezer teknőc definiálható és mozgatható. A teknőcálruhák színesen, harminckétszer harminckettes mátrixba rajzolhatók meg. Elforgatásukat, szerkesztésüket a MacLogoWriterhez hasonló funkciók segítik. Egy-egy álruhasor megszerkesztésével, a teknőc álruhájával együtt forgatható. A teknőc álruha más programból bevitt kép is lehet.

Az interaktivitás fokozására "készülékeket" hívhatunk elő a képernyőre, mint például vonalzó, amelyről leolvasható mennyit - vagy amelyen bejelölhető meddig - lépjen előre a teknőc, szögmérőt, vagy táblázatokat, skálákat, amelyből kiválaszthatók az utasítások paraméterei. Az interaktivitás fokozását jelenti, hogy amennyiben egy paraméter nélkül írunk

be egy parancsszót, akkor automatikusan megjelenik a megfelelő készülék, felajánlja a választást.

Az alkalmazható színek száma több ezer. A színválasztást is menük, skálák segítik. Az alkalmazható betűtípusok is változtathatók.

A pedagógusok régi jogos vágya teljesült: a technócöknök magyar szavakkal parancsolhatunk, az üzenetek, kérdések, menük magyarul jelennek meg. (A kis-Logo teljes Comlogóban futó változatán dolgozunk.)

A logo-pedagógia

A Logo nyelv és még inkább a logo-pedagógia sok olyan pedagógiai problémát képes megoldani, amelyekkel régóta küszködik az emberiség. Néhány pedagógiai irányzatot sorolok, egy értékes szakkönyv (4) nyomán, jelzem azok miben adtak jelentős fejlődést és vázolom azt, hogy hogyan segíti az ilyen irányú törekvéseket a Logo, illetve a logo-pedagógián alapuló módszerünk a Játékos Informatika..

Varga Lajos - Pék András egyik állítását igen fontosnak tartom. Egyetértek azzal, hogy *"az eddigi komputerrel segített oktatás legtöbb programja, illeszkedik a pedagógiai valósághoz, de nem adekvát önmagával."*

A logo-pedagógiát alkalmazó nem a régi tananyagot tanítja és nem a régi módon. Például a matematika területéről nem "újmatek"-ot tanítunk, hanem technóc-geometriát. Ezzel olyan részeit a matematikának, amit eddig nem, vagy legalábbis az adott korosztálynak megfelelő szinten nem tudtunk tanítani. Ismétlésekkel bemutatni a sokszögeket, az ismétlésekkor fellépő "hibával" bemutatni a spirálokat, szemléletes módja a geometriai struktúrák alakításának. Új módon mutatjuk be a szimmetriát (lásd Lego-Logo), szemléltetjük a véletlent, mutatjuk meg az osztók fogalmát. Ezen a területen említésre méltó Földes Petra (ÁKG) kolléganő munkássága.

Pavlov munkássága, a klasszikus kondicionálás elmélete kapcsán ezt olvashatjuk a hivatkozott műben: *"Az információtechnikai képzés kétségtelen előnye a kép-hang és mozgás, illetve mozdulat összekapcsolása. Ha azonban az előállított kép valóságból szerzett tapasztalatra nem támaszkodik, könnyen egyoldalú, egysíkú lesz a tapasztalat."*

Teljes mértékig egyetértek a megállapítással. A logo-pedagógia különösen a mi értelmezésünkben legfőbb originalitása és értéke az, hogy a számítógépet háttérbe szorítva, a számítógép mellett sőt helyett valós eszközöket (adó-vevő, robot, LEGO), tapasztalatszerző cselekedtetést (tégla-Logo, gyufa-Logo, informatikai eszközök gyártása, vizsgálata) csoportjátékokat (robot játék, Etesd!) használ.

Thorndike — behaviorista pszichológia — trial and error

Az általunk alkalmazott eszközök használata játékosan biztosítja az inger-reakció kapcsolatok megerősítését. A Logo-val vagy a LEGO-val való problémamegoldás során a próbálgatást részesítjük előnybe. (Mennyit kell lépnie a teknőcnek, hogy a kívánt pontba jusson? Próbáld ki! — Melyik szám jelenti a sárga színt? Próbálgatással keresd meg!)

... az instrumentális tanulás racionális és jól használható magva éppen ez: a tanulás feladatmegoldás és az egész személyiség akciója (ember esetében) továbbá kereső, heurisztikus tevékenység."

A Logo, a LEGO építés, a video felvételek készítése, a számítógépes írás és a számítógépes szövegszerkesztés, az "informatek" (ezek mind a Játékos Informatika elemei) különösen sok lehetőséget adnak a belátás útján való tanulásra.

A drive-redukció elmélete: *"Ami az információtechnikai képzést illeti, e téren a drive szint, azaz a tanulók motiváltsága kifejezetten jónak tekinthető. Ezt az aktivációs szintet fenntartani csak kifejezetten az újdonság varázsára hagyatkozva aligha lehet."*

A JI sokféle vonzó eszközt használ. Az aktivációs szintet a Logo rajzorientáltsága és "Rubik" hatása biztosítja. Az eszközök rendszert alkotnak. A LEGO konstrukciókat a számítógéphez kötjük. A számítógépes szövegszerkesztővel írjuk meg a video klippek forgatókönyvét stb.

Skinner — kis lépések, megerősítés — *"... bár a komputerre készített oktatóprogramok többsége felhasználja SKINNER alapgondolatait, ma már bátrabban kell előbbre lépniük, s a kislépések elvét egy konkrét, de nagyobb, problémamegoldással társított, önálló, alkotó gondolkodást kívánó vezérléssel kell kiegészítenünk."*

Ezt valósította meg a Logo! Kezdetben minden utasítás eredménye látható. Később lehet lépésenként, lehet csomagoltan, eljárások írásával próbára tenni ötleteinket. A strukturáltság a Logo legfőbb jellemzője, pedagógiai értéke. A JI módszere, célja az egyéni munkahelyek összehangolt munkája, közös nagy projektek (iskolaujság, elektronikai berendezés, filmforgatás) létrehozása.

Alaklélektan *"alakzatok, egészek, struktúrák elsődlegességét emeli ki ... a hirtelen belátást intuíciót"*

A Logo rajzorientált, strukturált. A JI jellemzője az eszközökkel való munka. Az algoritmikus gondolkodás fejlesztése, a hasonlóság felismertetése is alapcélunk.

Wiener — matematikai tanulásméletek

A szabályozástechnikát a JI képes "levinni" az alsó tagozatba. Erre példát az első magyar Logo konferencia harmadik napja is szolgáltat.

Klix "A matematikai tanuláselméletek egyikének tekinthető az élőszervezet és a külső világ közötti információcsere rendszerelméleti, algoritmuselméleti és információelméleti közös szempontú modellje..."

A meghatározásnak szinte minden szava egyben a logo-pedagógia deskriptora is!

Mediációs tanuláselméletek. "...a passzív képernyőnézést kiegészítő feladatmegoldások, mozgással-írással, rajzzal, ábrák készítésével... A komputerrel segített oktatásban pedig a manipulatív tevékenységek válnak dominánssá."

A JI -ban odáig jutottunk, hogy a képernyőnézés vált kiegészítő mozzanattá a robotjáték, az Etesd ..., játék, a padlóteknőc mozgatása, a manipulatív foglalkozások után.

Varga - Pék: Pedagógia a számítógépek korában könyve a Logo-ról is állást foglal: "Amikor egy tanuló LOGO-val "játszva" egy sünit..." "... körbe futtat, szimbólumokkal ellátott nyomógombokat nyomogat, mégpedig egy előre meghatározott sorrendben. Nem a tanuló teszi meg a kört, hanem a süni.

Motorikus tanulási tapasztalata a nyomógombra vonatkozik, nem egyezik meg a saját testtel végzett reális körmozgással."

A JI a szintóniát teljesebben kihasználja. A JI-ban elsődleges a ROBOTJÁTÉK, a gyermekek saját mozgása, az algoritmus átélése. A JI-ban először a kisgyermek, utána társa, teszi meg a kört, majd csak ez után mozgatja a padlóteknőcöt és csak végezetül a képernyőn levő jelet! Avagy a téglalagónál a gyermek maga épít. A JI oktatása során így a motorikus tapasztalatok a kisgyerekek saját testével, az általa manipulált elemek mozgatásával kapcsolatosak.

Úgy vélem a Logora alapozó paradigmánk sok követelményt kielégít:

Báthory Zoltán (5) pedagógiánkban az alábbi konzekvenciák érvényesülését tartja fontosnak:

—"*... aktív és produktív iskolai tanulás előtérbe kerülése*"

A JI és benne különösen a számítógép használat fokozottan aktív és produktív. A gyerekek munkáikat mindenkor elvihetik, felhasználhatják.

—"*... a tanulás-tanítás csábító egyoldalságainak ... feloldása ...*"

A JI a verbalizmustól nagyon messze van. A számítógép és a többi információtechnikai eszköz kezelését elmagyarázni csaknem lehetetlen, bemutatás nélkül. A gyakorlás során a pedagógus rendszerint már "csak" szemlélő, hallgató, csak közönség.

—"*... problémamegoldásnak, a felfedezéssel tanulásnak, a belátásos tanulásnak ... integrálása*"

Ez a logo-pedagógia jelszava is.

—"*... a cselekvés*"

A gyakori eszközhasználat vagy a csoportjátékok szükségszerűen cselekvéssel járnak együtt.

—"*... valóságos kommunikációs helyzetek teremtése ...*"

A számítógéppel, a videóval a JI game-ek során a kommunikációs helyzetek nem megjártak, hanem tényleges szükségszerű cselekedetek.

— Mind Báthory, mind Zsolnai kiemeli a differenciálás fontosságát.

A JI folytonosan differenciál. Hét számítógépnél vagy játékeszközknél nem is volna lehetséges megvalósítani, hogy a gyerekek azonos munkát végezzenek.

A Logo és egyáltalán az informatika oktatása tehát visszahat a pedagógiára. Amint az előzőekben láthattuk a pedagógiai követelmények jelentős részét az információtechnika

fejlesztésével tudjuk megvalósítani, de az információtechnika megváltoztatja a pedagógiát is:

Az információk gépesített kezelése egyre jobban kényszerít arra, hogy tudáson ne az információk felidézését értsük, erre a gép is képes. Most már valóban nem tudnunk kell egy adatot, hanem azt kell tudnunk honnan, hogyan hívhatjuk azt elő. Nem az információra, hanem az információ keresés módjára kell emlékezni, azon "utakra, melyeken az ismeretek kútfejeihez juthatunk". (Kölcsey)

A feladatmegoldás helyett a problémamegoldás kerül előtérbe. Egyre gyakrabban nem kell a munkát elvégezni, azt megtervezni, kiadni, megparancsolni kell tudnunk. (Igen precízen, kemény nyelven.)

Az információk gépesített, gyors hatékony módjaival a tanulás módja egyre inkább lehet a kutatás, keresés, a kísérletezés.

A hibákhoz való viszonyulás szükségszerűen végre humánus lesz. A számítógéppel dolgozva, vagy a hangfelvételek / videó felvételek készítésekor hibázni szabad, ez természetes, emberek vagyunk. Azt kell megtanulnunk, hogy kijavítsuk hibáinkat. A program addig nem működik, amíg szintaktikailag, szemantikailag hibás. A helyesírást ellenőrző szövegszerkesztő mindaddig megakad egy szónál, amíg nem javítjuk azt ki.

A számonkérés módja is szükségszerűen változik. Egyre gyakoribb, hogy puskázni nemcsak lehet, de kötelező. A jó tanár olyat is kérdez, amelyre ő sem tudja a pontos választ. Egyre gyakoribb lehet, hogy a gyermek nem szép feleleteiért, hanem jó kérdéseiért kaphat kiváló minősítést.

Hivatkozott irodalom

1. Seymour Papert: Észrengés. A gyermeki gondolkodás titkos útjai. SZÁMALK. 1988.
2. Votisky Zsuzsa szerk.: A Logo programozási nyelv. Műszaki Könyvkiadó, Budapest, 1986. p. 317.
3. Stuur, Addo: Windows für Kids von 8 bis 88 (Holland kiadás: Windows voor Kinderen, International Thomson Publishing, Bonn - London - Stamford - Boston - New York - Toronto - Singapore - Tokyo - Sydney. Ára 39.80 DM.),
4. Varga Lajos - Pék András: Pedagógia a számítógépek korában. LSI Alkalmazástechnikai Tanácsadó Szolgálat, Budapest, 1988. p. 315.
5. Báthory Zoltán: Tanulók, iskolák - különbségek. Egy differenciális tanuláselmélet vázlata. Tankönyvkiadó, Budapest. 1992.

Not All Birds Are Turtles

Developing Teaching / Learning Environments in Initial Teacher Training at Comenius University

Kalas Ivan, Blaho Andrej

Department of Informatics Education
Comenius University
842 15 Bratislava, Slovakia
tel., fax.: + 42 7 724 826
E-mail: kalas@fmph.uniba.sk



*Some birds are turtles (multiple). Some are not.
Find them with your mouse.*

Abstract:

We characterise goals and atmosphere of reformed curricula for initial informatics teacher training at Comenius University. We believe that thanks to certain tradition in informatics education in Slovakia, thanks to experience of other European countries and deep enthusiasm of several teachers and educators, innovative application of project approach brings in new creative spirit. Designing, developing, and evaluating interactive teaching / learning environments in new implementation of Logo for Windows proves to attract students and surprisingly broadens traditional horizons of Logo world. We are experimenting with Comenius Logo for Windows as a development environment that meets the needs of children, students, teachers, and developers, and supports student-centred exploratory approach in (informatics) education.

1. A Mosaic of Educational Background (Concerning Informatics)

To better understand our conception of reformed informatics curricula at Comenius University, we put forward some short notes on history of informatics education in our country:

- informatics in higher secondary education (that is, for students of 15 - 18 years) referred and often still refers mainly to **programming**. Within strongly centralised educational system of 80-ties, informatics education was given rather high preferences. Influenced by "programming as the third literacy" opinion of the previous decade, it became nearly a kind of national sport,
- gaining programming skills proved to be highly attractive for secondary students; local and national programming competitions are held every year - our teams are getting excellent placing even in the most ambitious international competitions¹. Although, hand in hand with broad popularity of programming, it

¹ International Olympiad in Informatics: 1991 in Athens - first position, 1992 in Bonn - first position, 1993 in Mendoza, Argentina - first position; International Software Competition (for secondary schools), Singapore 1992 - second position; East European Regional ACM International Collegiate Programming Contest, Sofia 1992 - first position, 1993 - second position, Final of ACM ICPC, Indianapolis - 20th position, first European position.

has become a kind of myth and legend for younger students ("I want to program. I don't know what it is, but I want to do it. Let me do some serious programming ..."),

- informatics education is a separate subject (that is, not merged into other subjects in a cross curricular way). Teachers of informatics are (often) graduates of university teacher training in Mathematics and Informatics, more rarely in Physics and Informatics,
- as far as hardware is concerned, till the end of 80-ties the most widespread computers used at schools were 8-bit monochrome computers of very poor quality, unreliable, with no disk storage etc. And still, however troublesome those computers were, they taught a lot of young people and teachers to get much more than 100% out of them, they promoted original software development (Pascal compiler, Karel the Robot 3D environment, Lisp interpreter, miniLogo) and our early research in informatics education,
- concerning primary education, there is no regular informatics education at present, although - initialised by enthusiastic teachers, parents, and researchers - there are some experimental courses and a lot of free time activities in this direction. This situation gives us absolutely free hand in deciding what to do with children. As far as we are quite sure that in future primary informatics education will be put on a regular basis, we devote considerable space in our curricula to it (which will become obvious in the following chapter of this paper),
- in 1993, department of Informatics Education was established at Comenius University which "legalised" and "professionalised" activities of a group of people with traditional informatics background, who have for years been involved in primary and secondary informatics education research². We are now trying to anticipate future development of the role of informatics in schools and have started new curricula based on our strong believe that informatics is much broader subject than its traditional reduction to programming; for more inspiration see also [ACM, 93]. That is, it can and will contribute much more to education by developing general skills of abstraction, algorithmic thinking, logical mind, skills to organise information, to communicate it, to share it, to co-operate over information, and - which is by far not of the minor importance - to enjoy education, to develop aesthetic feelings, to have fun,
- new ambitious implementation of Logo, Comenius Logo for Windows³, see [Blaho, 93], [Kalas, 93], expresses our believe in a new, **project approach** in school informatics. The main objective of the implementation is to provide students and teachers with strong learning environment supporting active exploratory involvement. Keeping original pure Logo teaching / learning philosophy, we tried to introduce all new achievements of software world of 90-ties into it. We are offering new powerful computer activity environment to students and teachers who have already seen a lot of professional computer games with perfect animation and are often disappointed by simplicity of some educational software packages.

To summarise the mosaic: we feel we are living exciting time. We as children were enchanted by electric bulbs and binary arithmetic of huge "steam computers". Now we are trying to be in of step with new trends in informatics education.

2. Project Approach at Schools and in Initial Teacher Training

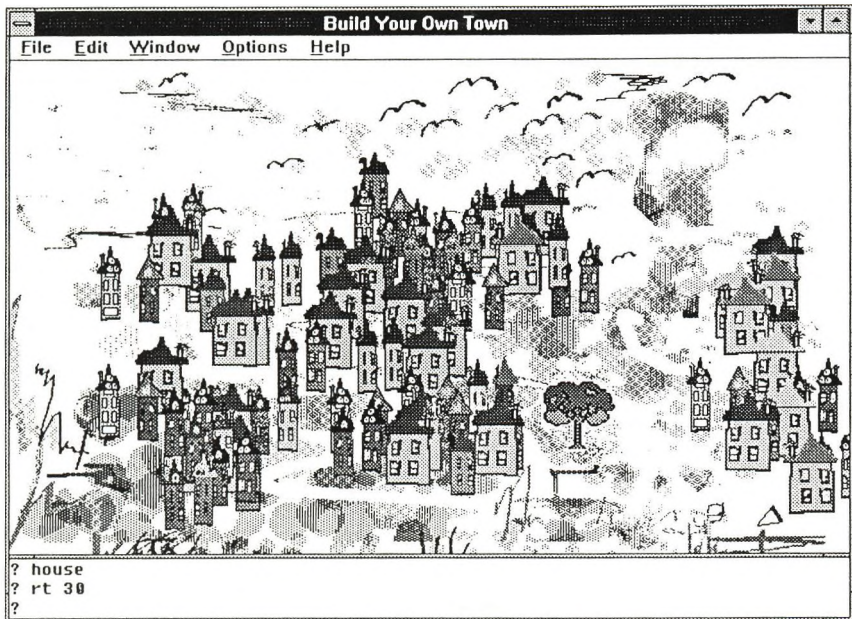
Out of the reformed curricula we want to present here in more detail a block of three courses closely tied together, which focuses mainly on informatics for primary and lower secondary education. (Higher secondary informatics education, as we have already put it in part 1, is much more programming-centred and in our curricula is covered more by another block of courses. By no means, however, are these two levels of education separated and separable in the initial teacher training. We believe that project approach will gradually attract more and more students and their teachers.)

Three courses of the block are:

- Symbolic Computations and Logo,
- Informatics in Primary and Lower Secondary Education (ages 6 - 14),
- Educational Software Design

² Experience of Western European countries has influenced our understanding of informatics education considerably. One of the authors had been involved in the Dutch-Czech-Slovakian Comenius Project aimed at introducing modern CIT into secondary education in 1991 - 93.

³ Titled SuperLogo voor Windows, this Logo environment is distributed in the Netherlands by A.W. Bruna Uitgevers B. V., Utrecht.



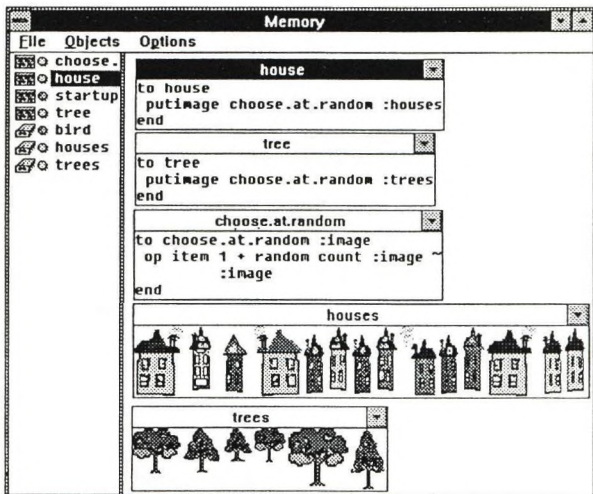
First of these courses profits considerably from the fact that students who take it, are already quite advanced programmers from their first university school year (and some of them partly from their secondary schools). Introductory intensive programming course of the first year concentrates on Pascal imperative programming, therefore we can afford to thrust them into a swimming-pool: Symbolic Computations and Logo course focuses mainly on Lisp-like data structures as lists and trees which are always present somewhere in the backyard of each non trivial application.

In the Logo environment we spend much time on recursion, recursive structures, recursive drawings, fractals, and polygons. From more traditional Logo turtle world we move ahead very soon towards some advanced features of the new implementation: multiple turtles, programming interactions by mouse, animation etc.

Our presumption is that an informatics teacher will find himself acting in many different professional roles: he will be involved in regular activities with his students (some of them will certainly be *computer hackers*); he will probably **develop** his own software applications⁴, he will **modify** (or at least dream of modifying) another applications to better fit his needs and needs of his students. He may as well **co-operate** with other non-informatics teachers to help them integrate information technologies into their subjects, he may act as a kind of *school informatics co-ordinator*, he might be asked to support *school administration*... In [Blaho, 93] we put forward our belief that powerful and appropriate (but not: powerful OR appropriate!) software environment may somehow unify all of these different roles and different needs: spanning from children-oriented computer world of a turtle to development tool for educational applications. In our reformed curricula we are trying to use Comenius Logo for Windows as such unifying environment.

⁴ In the Netherlands, for example, when visiting several Comenius Project schools, we met a lot of very enthusiastic teachers deeply interested in developing their own small applications and sharing them vividly with other colleagues throughout the region.

What are the main objectives of the block of three courses? Let us start from the other end: we will describe a simple environment, which we have been using in our experimental Informatics class for children ages 10 - 11 years. In this way we will illustrate our goals, conception, and methods used in the courses. Because this characterises the whole process, which our university students - future teachers - should go through.

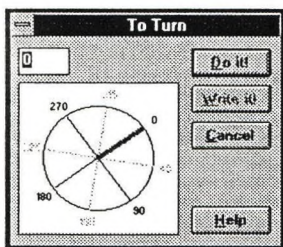


We are talking about **Build Your Own Town** project, which is an elementary introduction to Logo and its turtle geometry. Children just started their Informatics class a couple of weeks ago by 'Windows with your hands on the keyboard' method inspired by [Stuur, 93]. Children now fluently work with mouse, they are already able to produce their own *text-and-picture stories* in Paintbrush. They have already spent two lessons exploring Comenius Logo environment by going through our demo projects. They know that there is a turtle living at the screen. They know that it can walk around with its pen on or off (*penup* or *pendown*).

by themselves. They are building a town by making the turtle navigate through simple background picture with its *penup* using only basic *forward* and *right*, *left* commands. They are trying to investigate new features of Logo at the same time, and we are trying to make them find these features at the time when they really need them (not sooner). To build a house and plant a tree they use special commands *house* and *tree*, which put on the screen an image of randomly chosen house and tree at the actual position of the turtle.

Our goal is to make children explore the environment mainly

Typing in simple Logo commands like *forward*, *fd*, *right*, *rt*, or *house* is quite simple because Comenius Logo supports national language, that is, complete communication within its environment together with all commands of Logo language are realised in children's natural language (let it for now be English).



As every teacher using Logo knows, a lot of mistakes happen when children type inputs to commands⁵, like *forward 30* (abbreviated to *fd 30*). Even more complicated situation arises for children of this age when they need to express angles in degrees. We expected this but we let them try. In this way they discovered several **choosers** - they found out that it suffices to type in *right* and immediately press Enter. Instead of causing an error, **Turn Chooser** - a compass opens in which you can qualitatively (not measured in numbers) express how much the turtle should turn and in which direction. When you now press OK, the command with the proper input will appear in the command line (to see the complete instruction) and is executed. The same holds for *forward* (and as it will show up latter during their school year, nearly for all commands) - pressing Enter after *fd* command, a **Ruler** is displayed to help them say: go that much!

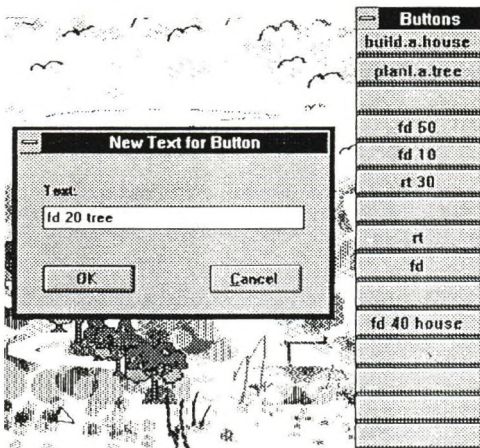
Now we showed another tool to children - **Buttons** with some pre-defined texts and associated behaviour. Instead of typing *house* and *tree* commands children start to choose buttons with *build a house* and *plant a tree* texts and press them by their mouse. However, quite soon they demand to use also some other buttons of the

⁵ More exactly, when they type Logo instruction, that is, name of a command together with its inputs.

Buttons window which were now empty. Yes, it is very easy, you just click on it with the right mouse button and type new text - any Logo instruction - in the dialog box **New Text for Button**. From now on, children do not need the text screen in this environment any more. Instead, their buttons will soon be occupied with commands like *fd 50* for long moves, *fd 10* for short moves, *rt 30* for "turning a little bit" etc. Now the work is much more effective for them.

It is also possible to define a button with incomplete instruction, for example *right* without any input. Such button opens **Turn Chooser** in which we specify the angle by dragging the hand of the compass.

Children are soon demanding more - some boys want to build roads among the houses. As far as we haven't planned such option, we start to improvise together with children: we define new buttons *setpenwidth*, *setpencolor*, *penup*, and *pendown*. To build a road now means to put turtle's pen down, press *setpencolor* button and choose a color from **Color chooser**, press *setpenwidth* button and choose pen width in its chooser, and start to move turtle by *fd* commands.



Some more tricks: follow the same road with white and narrower pen to add short lines in the middle. Define and use following button:

```
pd fd 9 pu fd 9
```

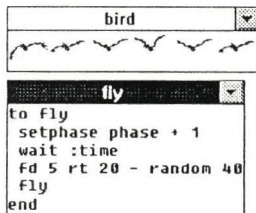
We believe that some small unexpected surprises or jokes should be present in any activity environment. As an example, we announced in the class that some of birds on the picture can fly. Children have to find them by themselves. First they type *find a bird* command (or define a button with the same text and press it), then they click on each bird: if it is the right one, it starts to fly. This is their first encounter with Comenius Logo animation.

How is this realised? Actually, more turtles can live at the graphics screen, not only one (their number is limited by 4000). Moreover, any of them can have its own **shape** which is always a series of small pictures (or phases as we call them). You can define and create your own images and assign them to any turtle.

Animation proves to be highly stimulating topic for children, although developing nice series of frames and making them move around the screen is not always simple. Therefore we prefer to start either using prepared libraries of such images which children will customise, colour, multiply etc. or to animate simple objects with simple movements, like a car or a plane, traffic lights changing their state etc. From developers point of view, animation is one of the most powerful tools: by animation, you can present children with the problem to be solved, by animation you can interpret their solutions, you can reveal the results and evaluation etc., see [Kalas. 93].

Some final remarks on **Build your own town** project: although it is very simple both from the developer's point of view and from the children's point of view, it is rather flexible. Children decide what they want to have there. The environment used for implementing such flexible and extensible environments must be also simple in its computational and educational philosophy (children are using the same environment as the developer) and still very powerful to support such effective activities without going deep into technical details.

We are aware of the fact that designing and implementing educational



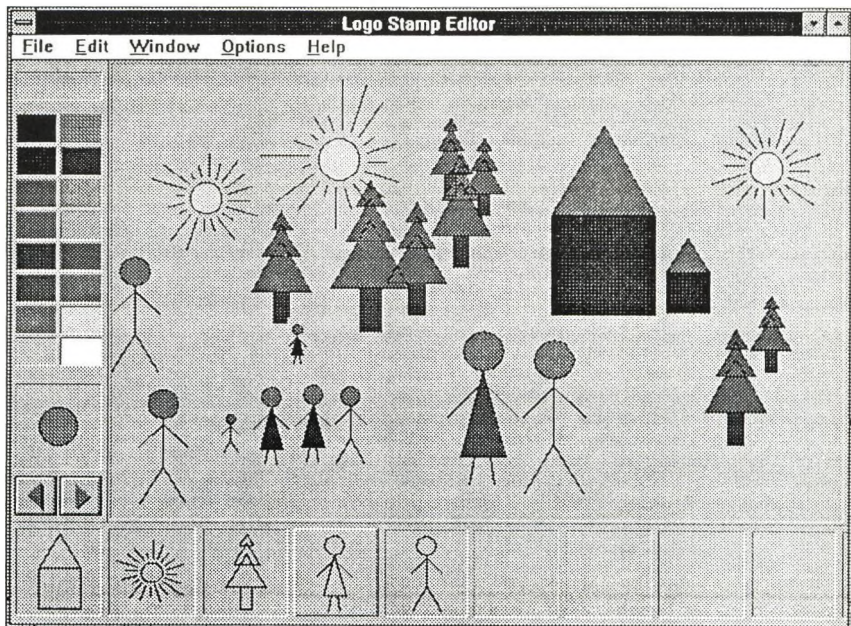
environments is only one side of the coin (maybe even the smaller one, if you can imagine such coin). Therefore we are planning to devote more than 60% of time in our block of three courses to educational aspects of the environments, discussing motivation and educational goals, planning and evaluating stages of development, and gaining practical experience with the 'end users'.

3. Another Example - Closer to Programming

In the **Build Your Own Town** project children navigate the turtle so that it places 'ready-made' houses and trees around the screen. To achieve this, they use basic turtle commands *forward*, *left*, and *right*. More of the traditional turtle geometry is present when they start to build a road going through the town. There are also other turtles moving above the town in disguise - crows, but these are only actors to be watched (or to give motivation to use Logo Image Editor and try some more animation). In total, the environment introduced in part 2 is creative, although more used by children than created. Still, there is one small but important exception to this classification: children are defining their own buttons, they are defining behaviour of these buttons and thus customising the environment. They choose a goal, a problem to be solved and develop their own tools to support its effective solution.

Thanks to the flexibility of underlying Logo environment, children can start with:

- associating simple **complete Logo command** (instruction) with a button, like *forward 50*,
- associating **incomplete Logo command** with a button, like *setpencolor*. Such command, however simple it may look, is less fixed and conceptually more complex. To press such button means to open a chooser and choose. In this way children develop the sense of inputs, and thus the introductory sense of variables,
- the sequence may go on by associating **several complete commands** with one button, for example to plant a row of trees by *fd 20 tree*. As we have seen, to draw white lines in the middle of the road, we can construct *pu fd 9 pd fd 9* button, that is, a kind of road machine which moves on without drawing by *pu fd 9*, then with drawing by *pd fd 9*. We can **repeat** the same action by clicking the button several times.



Complexity of the action associated with a button seems to be limited only by the size of the button. This we can really call a step towards abstraction: more complex action or a series of actions is expressed by one concept - clicking this particular button. In this moment we are not far from **defining our own command** (with no restrictions by the size of a button). In this way we are giving a name to certain complex action probably described by several Logo commands, like:

```
to next.tree
  fd 20
  tree
end
```

Now *next.tree* command can be typed in directly in the command line, or can be associated with a button. In this way children proceed to more and more complex abstractions assigning names to combinations of other names...

This is quite natural process for children in acquiring Logo world. Step by step they define very simple commands, like *square*, *rectangle*, *triangle*, they combine them into more complex ones: a *house* is a square with correctly placed *triangle*. Logo Stamp Editor environment, another example of more complex program developed in Comenius Logo, goes in this direction much further than **Build Your Own Town** project. Working with it, children define their own drawings, less or more simple, depending on their age, motivation, and abilities. Following is an example of a simple tree program:

```
to my.tree
  lt 90
  repeat 18 [fd 10 rt 20]
  lt 90
  fd 50 rt 90 fd 10 rt 90 fd 50
end
```

These drawings - a boy, a sun, a car, a tree etc., are integrated into the environment developed by a teacher. Each drawing (or better, the program which draws it) can be added by a child into a 'stamp-bar' at the bottom of the screen and later used in creating a composition. Children start to create their drawings in an *instructional* way, that is, by programming them in Logo language, and then create compositions in a *constructional* way - any stamp can be selected by a mouse, then used and coloured in the working area of Stamp Editor.

There is another interesting aspect in this environment: *shrinking and growing* of the stamps. Before using a stamp, we can select one of six possible sizes. When the turtle draws the stamp (that is, when it executes corresponding program), it makes "longer" or "shorter" steps according to selected size. Children will notice that "longer" steps of the turtle results in exactly the same drawing except that it is larger.



Logo Stamp Editor is interesting from the developer's and user's points of view in that both of them use exactly the same language: the developer has created an environment which is able to understand, accept, and process Logo program of a child, and incorporate it into the stamp bar. The child, that is, the stamp's developer, can later use these stamps for his / her compositions. The child here is neither using nor modifying the environment - he / she is co-creating it.

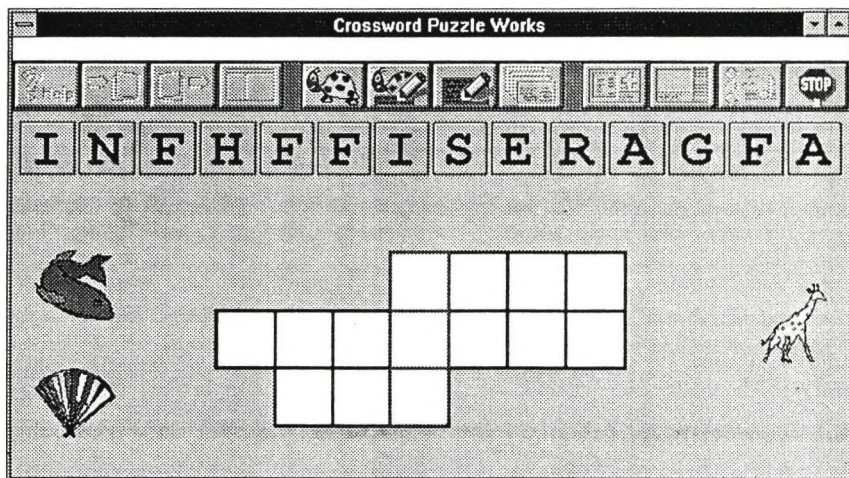
4. Early Results with New Curricula

Our reformed curricula are quite new. In the last winter term we went through the introductory course of the block - Symbolic Computations and Logo for the first time. The exam of the course required that each student developed an experimental small Logo environment. In the most cases students thought out their own topics and will spend much more time on them in the following courses.

Not all of these student projects were excellent, but several promise to result in very interesting environments. Here we are presenting one of them - **Crossword Puzzle Works**, in which children should fill in a crossword puzzle according to small pictures. They drag letters from the upper part of the screen into their correct positions. This project is interesting in that its author realised how many extensions the topic offers and decided

to go on considerably with its development. (The same happened with some other projects as well.) Her plan is to incorporate a feature, which we find to be very important for educational software: in the present implementation of the project all assignments are chosen at random from a fixed set of possible combinations. However, children want not only to solve assignments, but highly esteem the possibility to develop their own assignment to be later present to a friend as his assignment.

We hope that some of these student projects will be successful at schools and will play positive role in the current initial teacher training.



References

- [Stuur, 93] STUUR, A.: Windows Voor Kinderen, A.W. Bruna Informatica, Utrecht, 1993
- [ACM, 93] ACM Model High School Computer Science Curriculum, C. ACM, May 1993, Vol. 36, No.5. pp. 87 - 90
- [Blaho, 93] BLAHO, A., KALAS, I., TOMCSANYI, P.: Comenius Logo: Environment for Teachers and Environment for Learners, Supplement of EuroLOGO '93 Proc., Athens, 1993, pp. 1 - 11
- [Kalas, 93] KALAS, I., BLAHO, A.: Thomash the Clown's Circus: Order in Actions in Picture Languages. Informatics & Changes in Learning. Proc. of IFIP Open Conference, Gmunden, 1993, pp. XVII 1 - 4

