

SZAKDOLGOZAT

Virág Róbert

Debrecen
2008

Debreceni Egyetem
Informatika Kar

Weboktatás a középiskolában

Témavezető:
Dr. Juhász István
Egyetemi adjunktus

Készítette:
Virág Róbert
Matematika-Informatika

Debrecen
2008

Tartalomjegyzék

Bevezetés.....	4
Informatikai forradalom	5
Oktatás.....	5
Számítógépek az oktatásban.....	7
NAT.....	8
Kiemelt fejlesztési feladatok	8
Az informatikai műveltségi terület.....	9
Infokommunikáció	11
A tanulók sikeres motiválása.....	12
A motiváció	12
A motiváció idegéletteni alapjai.....	13
A külső és belső motivációkról iskolai megvilágításban.....	14
Külső motivációk.....	15
Belső motivációk	16
Eszközök az informatikában.....	21
Weboktatás területei	25
Alapvető hálózati ismeretek	26
Információ keresése az interneten	26
Információ elhelyezése	27
EditPlus 2.....	28
A HTML nyelv alapjai	30
A HTML tagok.....	34
Típushibák	39
Projektor használata.....	40
Kommunikáció az interneten.....	41
Összefoglalás.....	44
Mellékletek.....	47
1. melléklet	47
A C programozás alapjai	47
Forrás:.....	48
2. melléklet	52
Forrás:.....	52
3. melléklet	54
LINUX - UBUNTU	54
Forrás:.....	56
4. melléklet	60
Webcím.....	60
Forrás:.....	62
Irodalomjegyzék.....	67

Bevezetés

Szakedolgozatom fő témája a középiskolában történő weboktatás. Ez a terület nagyon fontos, hisz ma már egyre nehezebben tudjuk nélkülözni munkánk során az internetet. Szakedolgozatomat próbáltam úgy felépíteni, mely kiemeli azokat a területeket, amiket a középiskolában a tanulóknak el kell sajátítani. Munkám leginkább azoknak szól, akik a weboktatáshoz tartozó anyagrészeket szeretnék tanítani, ugyanis e témákat nem fejtem ki olyan szinten, hogy el lehessen sajátítani azokat, inkább próbáltam didaktikai szempontok szerint megközelíteni.

Munkám első részében egy kis betekintést teszek az informatikai forradalom napjainkra gyakorolt hatására, az oktatásban való megjelenésére. Megvizsgálom, hogyan jelenik meg napjainkban maga az informatika, mit mond a NAT az informatika oktatásáról, azon belül is azon fejlesztési feladatokról, melyek a weboktatáshoz kapcsolódnak.

Dolgozatomban külön fejezetet szánok a tanulók motiválásának, melyben megpróbálom a külső és belső motivációk segítségével a sikeres motivációt körbejárni. Ebben a fejezetben törekedtem általánosan írni a témáról, így az nem pusztán az informatika tantárgy keretein belül lehet alkalmazni.

Ezt követően szemügyre veszem azon eszközöket, melyek használata nélkülözhetetlen a hatékony informatikaoktatásban. Külön foglalkozok majd egy ma még nem igazán elterjed eszközzel, az aktív táblával.

Dolgozatom következő részében részletesen végigtárgyalom a weboktatás különböző területeit, s azok tanításának módszereit. E részben az alábbi területekkel foglalkozom részletesebben: alapvető hálózati ismeretek, melyek alap elsajátítása ma már majdhogynem nélkülözhetetlen, információ hatékony, céltudatos keresése az interneten, információ elhelyezése és nem utolsósorban, a legelterjedtebb internetes kommunikáció formáinak és azok szabályainak megismerése.

Informatikai forradalom

Az utóbbi néhány évtizedben számos olyan társadalomelméleti mű jelent meg, ami kísérletet tett korunk legfontosabbnak vélt társadalmi átalakulásának megragadására. A legtöbb szakember a "modern társadalmi változások" lényegének az "információs társadalom" kialakulását tartja. A változás tétje és ígérete egy a beszéd és írás kultúrtörténeti szerepével egyenértékűnek látszó új információkezelő minőség létrejötte. Az integrációs tendenciák fokozatosan felgyorsulnak. Az ezredvég a telekommunikáció, a számítástechnika, az irodatechnika, az információipar- és kereskedelem, a tudomány, az oktatás, a nyilvánosságcsatornák és a privát szféra grandiózus összeolvadását ígéri. Az emberi evolúció minden eddiginél nagyobb mértékben válik "extraszomatikussá", tárgyakba/eszközökbe helyezetté. A legjelentősebb ezek közül a számítógép, az az eszköz, ami képes egységes rendszerbe integrálni az írás, a hang, a (mozgó) kép sokszorosítását, őrzését és továbbítását, valamint az adott információs tevékenységekre épülő elkülönült gazdasági-társadalmi intézményeket is. A hangképző szervek és a toll után a harmadik minimálrendszer "fizikai hordozójaként" vélhetőleg egy multifunkciós, mobil, hálózatra kapcsolt, számítógép alapú személyi készülék áll a mindennapos tevékenység szolgálatába az időszervezés, a tanulás, a megismerés és a kommunikáció új médiumaként, egyesítve az információszerzés, termelés, feldolgozás, továbbítás, tárolás, egységesítés és visszakeresés funkcióit.

Az informatikai forradalom kiteljesedése nem csak globalizál, hatékonyságot növel, de egyben felértékeli a tudásipar, és abban a kutató és a fejlesztésre képes értelmiség szerepét. Átalakul a tudományos tevékenységgel szembeni napi társadalmi elvárás: a tudós továbbra is feltaláló - feltaláló kreativitását a bonyolult rendszerek szüntelen továbbfejlesztése mind jobban igényli. Az informatikai forradalom bonyolult rendszereket áttekinteni képes, a helyi innovációra - fejlesztésre orientált szakértelmiséget kíván.

Oktatás

Az évszázados iskola-minta, amelynek jó értelemben vett konzervativizmusa az értékek és műveltségi alapok változatlan megőrzése mellett stabil tartalmat és stabil módszereket jelentett, végérvényesen elavult. Alapvetően megváltozik a végzett diákokkal szemben támasztott követelmény: a "mi került a fejébe" helyett a "mire képes" a mérce. A jövőben a

tanároknak sokkal nagyobb mértékben kell összpontosítaniuk a diákban zajló motoros, verbális és mentális folyamatokra. Változik a műveltségzmény: "...a mai értelemben vett művelt ember az, aki képes megérteni, értelmezni és felhasználni a környezetében lévő hatások óriási tömegét. A műveltség szélesebb körű meghatározásának magába kell foglalnia a szimbólumok megtanulását, az üzenetközvetítőket, a nem verbális nyelvet, a kommunikációs csatornákat és effektusok emberi viselkedésre gyakorolt hatását." Mindez konkrét képességekre is lefordítható - Hechey szerint ezek a következők:

1. Döntések és tendenciák következményeinek előre látása (extrapoláció)
2. A jövő alternatív modelljeinek elgondolása ill. megalkotása
3. Holisztikus gondolkodás, rendszerelmélet
4. Az információ kiválasztásának, megszerzésének és csökkentésének képessége
5. A többértelműség és az ellentmondások tolerálása
6. Különböző közegek (média) útján történő kommunikálás, valamely médiumban (pl. szöveg) megjelenő közlés "lefordítása" egy másikra (pl. film).

A számítógép-vonzáspont köré kiépülő új készségek azonban nem önmagukban állnak: szükségszerűen a maguk képére formálják a három hagyományos alapkészséget. Az olvasás a szövegen tologatott optikai felismerő kártya segítségével gyorsabban megtanulható, s egy idő után már csak a problémás szavakra kell használni. A kézírás mellé a billentyűzetkezelés lép, a javítóprogramok miatt gyorsabb helyesírás tanulóval, kevesebb hagyományos írásproduktóval, mind több "szövegszerkesztett" anyaggal. A számolási feladatoknál használt zsebszámológépről már korábban is bebizonyosodott, hogy hatékonyabb tanulást, jobb megértést, nagyobb vonzalmat eredményez. E három alapkészségnek ki kell egészülnie másik 3-mal, egy a vizuális formanyelv kezelését lehetővé tevővel, egy az információkörnyezettel való kétirányú kapcsolattartást biztosító kommunikációs készséggel, és egy a megszerzett információk helyes és célirányos feldolgozását rutinszerűen lehetővé tevő gondolati-logikai szerkezetek birtoklását jelentő alapkészséggel. Mindez azt is jelenti, hogy nem csak a tanítás tartalmát, hanem annak technológiáját is meg kell újítani - a tekintélyvel alapuló verbális tanítás helyett, ami az információreproduktó és átadás alapját képezte, olyan kell, amely kreatív gondolkodásra és problémamegoldásra, s a kiterjedt párhuzamos oktatási lehetőségek igénybevételenek megtanítására épít.

Számítógépek az oktatásban

A számítógépek iskolai megjelenésének első időszakában a hagyományos tantárgyak oktatását megkönnyítő új eszközként tekintettek rájuk. Kiepült a "programozott oktatás" teljes metodikája. A számítógép többféle lehetőséget, "üzemmódot" kínált a pedagógusoknak: irányító mód, ismétlő és gyakorló mód, tudományos modellek képi megjelenítése, oktató játékok, stb. Az eszközként való megjelenés egybeesett a programozási nyelvek tanításának megkezdésével, s a programozás ridegnek, bonyolultnak, elvontnak érzett világa elidegenítő effektusként hatott. Ellenérzéseket váltottak ki az olyan kijelentések, mint "a számítógép-programozás néhány éven belül olyan lesz, mint egy második ábécé", a "géptanárokkal" kapcsolatos zavaros prófécíák vagy a gépi oktatás túlzásba viteléből fakadó szélsőségek. A számítógépek árának csökkenésével beköszöntő tömeges használat nyomán számtalan vizsgálat igyekezett tisztázni a gépek valódi szerepét, hatását és lehetőségeit. Ezek egyértelműen igazolták, hogy a számítógép használat javítja a gyerekek ismeretbefogadását, egyénivé teszi a tanulási folyamatot. A számítógép használók fogékonyabbak a matematikára és feladatmegoldó készségeik javulnak. Az iskolában való használatuk alapvetően három összetevőjű: számítógéppel segített tanulás, gondolkodási, algoritmizáló képesség fejlesztése, számítógépes műveltség megalapozása.

NAT

Az eddig leírtakkal összefüggésben vizsgáljuk meg a magyarországi helyzetet, ezen belül is a közoktatás tartalmi szabályozásának legmagasabb szintű alapidokumentumát, a NAT-ot, melynek célja a közoktatás szemléletének megalapozása, azaz minden magyar iskolára érvényes általános célrendszer meghatározása.

A dokumentum figyelmet fordít az emberiség előtt álló közös problémákra. A NAT-ban képviselt értékek, az egységes, alapvető követelmények és az ezekre épülő differenciálás egyaránt azt a célt szolgálják, hogy a tanulók - adottságaikkal, fejlődésükkel, iskolai és iskolán kívüli tanulásukkal, egyéb tevékenységeikkel, szervezett és spontán tapasztalataikkal összhangban - minél teljesebben bontakoztathassák ki személyiségüket. A NAT a tanulók képességeinek fejlődéséhez szükséges követelmények meghatározásával ösztönzi a személyiségfejlesztő oktatást. Ez azonban csak úgy lehet eredményes, ha az intézmények pedagógiai programja, nevelési, tanítási-tanulási folyamata teret ad a színes, sokoldalú iskolai életnek, a tanulásnak, a játéknak, a munkának. A NAT olyan pedagógiai tevékenységet feltételez, amelyben a tanulók tudásának, képességeinek, egész személyiségének fejlődése, fejlesztése áll a középpontban.

A Nemzeti alaptanterv három részből épül fel, tartalmazza a műveltségi területeket, tartalmi szakaszolásokat és a fejlesztési feladatokat.

A NAT mellett elengedhetetlenül szükséges a kerettantervek elkészítése, mivel egyrészt ezeken keresztül valósulnak meg az egyes műveltségterületek építő tantárgyainak szabályozása a helyi tantervekben, valamint ezek szabályozzák a tartalmi egységet, és teszik lehetővé az iskolák közötti átjárhatóságot.

A Nemzeti alaptanterv tartalmi területeken önállóságot biztosít az iskoláknak, akiknek módjukban áll helyi tanterveikhez átvenni egy, a Nemzeti alaptanterv elfogadása után elkészített kerettantervet, vagy összeállíthatják azt több oktatási program alkalmazásával.

Kiemelt fejlesztési feladatok

A Nemzeti alaptanterv kiemelt fejlesztési feladatai közé tartozik a helyes énkép, önismeret kialakítása, a hon- és népismeret, az európai azonosságtudat – egyetemes kultúra, az aktív állampolgárságra, demokráciára nevelés, a gazdasági nevelés, a környezettudatosságra nevelés, a tanulás tanítása, a testi és lelki egészség, a felnőtt lét szerepeire történő felkészítés.

Ezen „kiemelt fejlesztési feladatok kulcskompetenciákra épülnek. Összekötik a műveltségterületek bevezetőit és fejlesztési feladatait”. Vagyis az informatika, mint műveltségterület által kibővíthetjük a fenti felsorolást, a hozzá tartozó fejlesztési feladatokkal. Nézzük meg az informatika műveltségterületet konkrétabban, és vizsgáljuk meg e fejlesztési feladatokat közelebbről, valamint célirányosabban.

Ilyen kiemelt fejlesztési feladatnak tekinthetjük az információs és kommunikációs kultúra kialakítását is.

Az informatikai műveltségi terület

A nemzeti alaptanterv alapelvei és céljai között fontos szerepet kap az információ társadalmi szerepe. Ez egyáltalán nem meglepő hiszen „az egyén érdeke, hogy időben hozzájusson a munkájához, életvitele alakításához szükséges információkhoz, képes legyen azokat céljának megfelelően feldolgozni és alkalmazni”. Éppen ezért a NAT célként jelöli meg a megfelelő információszerzési, -feldolgozási, adattárolási, -szervezési és -átadási technikák elsajátítását, valamint az információkezelés jogi és etikai szabályainak az ismeretét.

Fontos, hogy e technikák, valamint ismeretek mindig naprakészek legyenek, leginkább azért, mivel olyan területről van szó ahol nagyfokú a technológiai ismeretek fejlődése. Ezt nap mint nap tapasztalhatjuk az egyre növekvő intelligens szolgáltatások számában, valamint azáltal, hogy a különböző szolgáltatások, „rendszerek egyre szélesebb körben teszik lehetővé a felhasználói beavatkozást.” Éppen ezért a célok között szerepel, „hogy a tanuló figyelmet fordítson informatikai ismereteinek folyamatos megújítására.”

A nemzeti alaptanterv alapelvei, céljai között említést kap még ezen kívül a földrajzi elhelyezkedésből és az anyagi lehetőségek különbözőségéből adódó esélyegyenlőtlenségek csökkentése, valamint az információ nyilvánossá és mindenki számára hozzáférhetővé válása által a demokrácia erősítése.

A nemzeti alaptanterv az informatikai műveltségi terület kapcsán szót ejt, a pedagógus megváltozott szerepköréről vagyis, hogy „az ismeretátadó és számonkérő pedagógusból az ismeretek közötti eligazodást segítő, tanácsadó, az információt értékelni, abban kételkedni tudó tanulók nevelőjévé válik”. Változik ezen kívül az iskola, mint szervezet szerepe is. Fontos célként jelöli meg az önálló ismeretszerzés elérése érdekében – a könyvtárakhoz hasonlóan – a számítógépteremben lévő eszközökhöz való hozzáférést a tanórákon és azokon

kívül is. Végül pedig felhívja a figyelmet a többi műveltségterülethez való kapcsolódás fontosságára.

Az informatika műveltségterület fejlesztési feladatai:

1. Az informatikai eszközök használata
2. Informatika-alkalmazói ismeretek
 - 2.1 A gyakori életben használt legfontosabb írásos formátumok gépi megvalósítása, igény a mondanivaló lényegét tükröző esztétikus külalak kialakítására
 - 2.2 Adatbázisok, adattáblák alkalmazása, keresés az adatbázisban
3. Infotechnológia (problémamegoldás informatikai eszközökkel és módszerekkel)
 - 3.1 Az adott probléma megoldásához szükséges módszerek és eszközök kiválasztása
 - 3.2 Algoritmizálás, adatmodellezés (a hétköznapi életben és az iskolában előforduló tevékenységek algoritmizálható részleteinek felismerése és különféle formákban történő megfogalmazása)
 - 3.3 Egyszerűbb folyamatok modellezése, a paraméterek módosítása
4. Infokommunikáció
 - 4.1 Információkeresés, információközlés
 - 4.2 Információs technológián alapuló kommunikációs formák
5. Médiainformatika
6. Az információs társadalom
7. Könyvtári informatika
8. Az elektronikus vásárlás szerepe a XXI. században

Most szűkítsük a kört, és nézzük meg magán az informatika, mint műveltségterületen belül a dolgozat fő témáját, a weboktatással foglalkozó, valamint e területet lefedő részeket. E témakört majdnem teljes egészében lefedi az infokommunikációról, valamint az elektronikus vásárlás szerepéről szóló rész, de azért más fejlesztési feladatoknál némi átfedést láthatunk. Ilyenek például a médiainformatika, az információs társadalom és a könyvtári informatika.

Infokommunikáció

Az infokommunikáció az internet széles körű elterjedésével és mindenki számára – viszonylag könnyű – hozzáférhetőségével vált fontos, a tanulók által mindenképpen elsajátítandó fejlesztési feladattá. Arról nem is beszélve, hogy e terület milyen rohamosan fejlődik napjainkban, így mindenképpen fontos lépést tartani e területtel.

A nemzeti alaptanterv az internetet, és annak megismerését olyannyira fontosnak tartja, hogy külön fejlesztési feladatként jelenik meg, és e fejlesztési feladat már általános iskola első osztályában megjelenik. A középiskolába kerülő diákok tudása, az egyes általános iskolák felszereltségének, különböző tanítási módszereknek köszönhetően igencsak eltérő. Ezért is fontos az infokommunikációról szóló ismeretek átisméltése, azonos szintre hozása, valamint tovább mélyítése, látókörük kiszélesítése. E fejlesztési feladat egyik kiemelten fontos célja, hogy a diákokat megtanítsa az interneten való hatékony, céltudatos információszerzésre, valamint az internetre történő információ elhelyezése. Lényeges tehát az információ tudatos elrendezése a későbbi felhasználás elősegítése érdekében, valamint a hatékony információszerzési technikák kidolgozása. Másik fontos cél a másokkal való két – elektronikus levelezés általában, személyek közötti audió- és videotelefonálás – és a sokoldalú – levelezőlisták, csevegőcsatornák, videokonferencia-rendszerek – kommunikáció elsajátítása.

A sikeres, eredményekben gazdag oktatáshoz elengedhetetlen a tanulók megfelelő motiválása, ezért vizsgáljuk meg mélyebben e témát.

A tanulók sikeres motiválása

Mindig is erőteljes elméleti és gyakorlati törekvés élt a tanulók tanuláshoz való pozitív viszonyának kialakítására és ennek érdekében a szükséges motivációk létrehozására. Dolgozatom következő részében magával a motivációval, annak fogalmával, idegéletti alapjaival foglalkozom, majd a motivációk sokaságát megpróbálom két, egymással szembeállítható csoportba sorolni, ám mindezt nem pszichológiai, hanem amennyire csak lehet már iskolai megvilágosításból. Mindeközben célom, nem csupán a motiváció, hanem a sikeres motiváció megragadása, hisz függetlenül attól, hogy a tanárok mennyire motiváló típusúak a középiskolában, a gyerekek mindig is motiváltak lesznek. Ezért a legfontosabb dolog, hogy a tanárok részéről jövő leginkább tanulásra irányuló motiváció sikeres legyen. Ebben a vonatkozásban nyugodtan kijelenthetjük, hogy a siker záloga, hogy a tanárok részéről jövő motiváció „erősebb”, hatásosabb legyen az egyéb, más területekről érkező – általában talán zavaró, hátráltató – motivációknál. Beláthatjuk, hogy ez a mai világban nagyon nehéz dolog, hisz megszámlálhatatlanok azok a dolgok, amelyekkel a mai diákság szembenéz, vagy amelyek elterelik a figyelmüket.

Sajnos, nagyon nehéz konkrétan meghatározni mi is a sikeres motiváció titka, ugyanis nincs konkrét módszer, nincs konkrét algoritmus, azaz eljárás, mely minden helyzetben helyt állna, vagy akárcsak egy adott helyzetben minden diáknál, diáktípusnál működne. Mint ahogy minden ember különbözik, úgy minden diák is más-más preferencia rendszer alapján dönti el, hogy mikor mit, és főleg miért tegyen meg – s talán itt a miért kérdésben már felfedezhetjük magát a motivációt –, mások a befolyásoló tényezők, melyek hatással lehetnek – s vannak is – döntéseikre. Ezért amennyire csak lehet, saját középiskolai tapasztalataim, valamint ismerőseimmel, barátaimmal folytatott beszélgetések segítségével próbálom majd körüljárni, megközelíteni a sikeres motivációt.

A motiváció

A motiváció kifejezés a latin *movere* (mozogni) szóból ered. A motiváció az a pszichés állapot, ami közvetlenül vagy közvetve cselekvésre ösztönöz.

Mikor alakul ki motiváció, illetve van-e olyan pillanat az ember életében, amikor éppen nem motivált valamire? A válasz egyértelműen nemleges. A fogamzás pillanatától a halál beálltáig mindig motivált állapotban vagyunk. Úgy is fogalmazhatnánk, hogy a motiváltság az

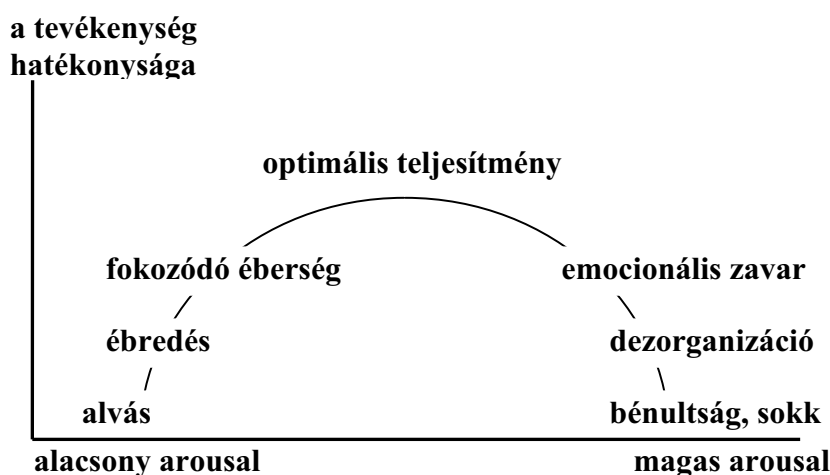
élő szervezetnek egyik legfontosabb sajátja. Ennek bizonyítására és jobb megértésére tekintünk át a motiváció idegrendszeri történéseit.

A motiváció idegéletteni alapjai

A motivált állapot kialakulása szoros kapcsolatban van az idegrendszer működésével. Ezen belül is kiemelt jelentőségű a hálózatos idegrendszer szerepe. Ez az agyvelőben található, szerkezeti felépítése pókhálószerű.

Működését tekintve aspecifikus, ami azt jelenti, hogy nincsenek olyan speciális ingerek, melyekhez működése kötött. Amíg pl. a halló idegpálya csak hangingerek, a látó idegpálya csak fényingerek hatására kerül ingerületi állapotba, addig a hálózatos idegrendszer számára minden inger hasznosítható. Amolyan „mindenevő” módjára bármely ingertől ingerületi állapotba kerülhet, s hálózatos elhelyezkedése révén hol felerősíti, hol gátolja a specifikus idegpályák működését. Jelzéseit a kéreg alatti központokon keresztül az agykéregnek is leadja, így jön létre az úgynevezett „háttér izgalmi-szint”, azaz arousal-szint. Az arousal-szint tehát a szervezet éberségi szintje, amely a mély alvást jellemző igen alacsony szinttől az erősen túlingerelt állapotig sok-sok árnyalatot tartalmazhat.

Az idegrendszeri arousal-szint és a teljesítményre való képesség összefüggését ábrázolja a következő függvény.



Létezik az arousalnak egy olyan optimális izgalmi állapota, melyhez optimális teljesítmény és pozitív emocionális élmény társul. Az ettől való eltérés – akár pozitív, akár negatív irányba – rontja a teljesítményt, és fokozza az érzelmek negatív töltését. Ilyen esetekben arra vagyunk motiváltak, hogy arousal-szintünket ismét optimális övezetbe juttassuk.

A nem specifikus hálózatos idegrendszer mindig működik, így éberségi szintünk soha nem nulla. Ez magyarázza azt, hogy élő szervezetben a motivált állapot folyamatos. A motiváció kialakulásában természetesen részt vesznek a specifikus érző és mozgatópályák is, de csak ezek működése nem biztosítaná a folyamatosságot.

Mindezek után térjünk rá fő témánk egyik részére a motivációra, s nézzük meg azt iskolai megvilágításban. Mint azt látni fogjuk a következőkben a motivációt két egymástól jól elkülöníthető csoportra tudjuk osztani. A külső, valamint a belső motivációkra. Vizsgáljuk meg most ezeket tüzetesebben.

A külső és belső motivációkról iskolai megvilágításban

A motiváció belső, magatartást szabályozó folyamat, minden tevékenység kiindulópontja, irányítója, vagy befejezője. A tanulás fő mozgatórugója: a jövőre, a felnőttkori munkatevékenységben betöltött szerepre való felkészülés. A motívum, illetve a motiváció gyűjtőfogalmat jelent, minden belső cselekvésre, viselkedésre készítő tényezőt magába foglal. Az emberek életében motívumok bonyolult rendszere működik és hat. Túlnyomó többségük nem vele született, hanem már az élet folyamán módosult, vagy újonnan elsajátított. Az egyéni élet során szerzett tapasztalatok a tanulásban is döntő tényezőt jelentenek, a motiváció állandóan változik, fejlődik – a motívumokat is „tanuljuk”. Ez többféle módon történhet. Motivált állapotban az érzékenységünk fokozódik a megfelelő céltárggyal szemben. A motívum akkor aktív, amikor valamilyen belső hatás vagy külső körülmény működésbe hozza. Beszélhetünk belső és külső motivációról. Külső motiváció lehet osztályzat, szülői elvárás, a tanár személyisége. Belső motiváció lehet tudásvágy, sikerorientáció, kudarcból való félelem, mások elismerése, célkitűzés. *Claparčde* (1873–1941) felismerte a belső motiváció fontosságát: a hallgató ne azért dolgozzék, hogy másvalakinek engedelmesskedjék, hanem azért, mert vágyat érez a munka elvégzésére. A külső motiváció beépülhet a belső motivációba: a belső motivációhoz hozzájárul a külső elvárásokhoz való alkalmazkodás.

A külső és belső motiváció ezen megközelítése fontos tényező ahhoz, hogy minél közelebb kerüljünk a sikeres motivációhoz. Vizsgáljuk meg először a külső motivációkat, hogy azok milyen módon, milyen mértékben befolyásolják a diákok cselekedeteit, döntéseit.

Külső motivációk

Azt nem nehéz belátni, hogy maga az osztályzat nem feltétlenül a tanártól függ – legalábbis nem csak a tanártól. A jegyeiktől az ember elvárja hogy tárgyszerűek legyenek (az objektivitás követelménye), vagyis azoknak nem szabad függeniük az értékelő, adott esetben a tanár személyétől. Nem tükrözhetik a tanár hangulatát, előítéleteit, beállítódásait, érzelmeit. Ezen kívül az értékelésnek megbízhatónak kell lennie (reliabilitás), azaz ugyanarra a tudásra mindig ugyanazt az osztályzatot kell a tanulónak kapnia. Végül az osztályzatoknak ki kell elégíteniük az érvényesség (validitás) követelményeit, azaz a jegyekben annak a tudásnak kell tükröződnie, amelynek a jellemzésre az adott osztályzatot használni akarjuk. Ha az osztályzat mind a három általunk támasztott követelménynek megfelel, akkor beláthatjuk, hogy bár az osztályzatot a tanár adja, mégis, mint külső motivációt maga a diák befolyásolja. (Természetesen a valóságban sajnos sokszor nem ez a tendencia.)

A szülői elvárás még bonyolultabb, összetettebb, és a tanári oldalról sok esetben – és ez a szülőktől függ – még inkább megfoghatatlan dolog. A tapasztalatok azt mutatják, hogy azok a tanulók, akik az iskolában jól teljesítenek általában a szüleikkel is jó viszonyban vannak. S e tanulók szülei jobban tartják a kapcsolatot a tanárokkal akár szülői értekezlet, akár fogadóóra segítségével. Másrészt, sajnos vannak esetek, amikor vagy a szülő nem törődik eléggé a gyermeke iskolai eredményeivel, tanulmányaival, vagy maga a diák nem érzi eléggé motiválva magát a szülő részéről. Mind az előbbi, mind pedig az utóbbi esetben a tanár nem sok mindent tehet a helyzet javítására.

A tanár személyisége talán az egyik legösszetettebb dolog, ugyanis mindamelllett, hogy egy bizonyos mértékig formálható, ezzel szemben bármilyen kísérlet ennek megváltoztatására a tanár részéről értelmetlen, és egyben felesleges dolog lenne. Egyrészt azért, mert ezt nem várhatjuk el egy tanártól, mivel neki kell nevelnie, formálnia a diákokat és nem fordítva. Valamint az elképzelhetetlen, hogy a tanár személyisége minden diáknak egyaránt megfeleljen.

Egy pedagógus nem igazán tudja közvetlenül a külső motivációkat befolyásolni, sok esetben ezek a külső motivációk sem feltétlenül elegendőek arra, hogy sikeresen motiválva legyenek a diákok.

Valószínűsíthetjük, hogy a választ arra a kérdésre, hogy hogyan tudjuk sikeresen motiválni a középiskolás diákokat a belső motivációk között találjuk meg. Ezek a belső motivációk,

mint ahogy azt már előzőleg felsoroltuk a tudásvágy, sikerorientáció, kudarcból való félelem, mások elismerése, valamint célkitűzés. Vizsgáljuk meg most ezeket is.

Belső motivációk

A tudásvágy, mint a diákokat belülről motiváló dolog, már jó kiindulópont célunk – a sikeres motiváció keresése – eléréséhez. A baj talán csak az lehet, hogy általában ennek a tudásvágnak a középpontjában nem azok a dolgok, tudásanyagok, tárgyak állnak, melyet a tanár szeretne elsajátítani a diákokkal. Feltehetjük a kérdést, hogy akkor milyen dolgokra irányul ez a tudásvágy? A válasz egyáltalán nem bonyolult, elég, ha egy kicsit magunkba nézünk, s felidézzük középiskolásban töltött éveinket. Egy diák azon „szükségletei”, hogy még több mindent tudjon meg bizonyos dologról, dolgokról, leginkább hobbiszinten űzött tevékenységei köré összpontosulnak.

A sikerorientáció talán szerencsésebb dolog abból a szempontból, hogy valamelyest – s ezt megint diákja válogatja – már az iskolai tanulásban is szerepet kap. Gondolhatunk itt egy felmérő megírására. Ebben az esetben mondhatjuk, hogy a diákot a sikerorientáltság vezérli, amikor megpróbál akár csak egy minimális teljesítményt elérni – legyen az akár egy elégséges. Hisz itt most nekünk nem szándékunk eldönteni, hogy melyik diáknak milyen teljesítmény számít sikernek. Azt azért szigorúan hozzá kell tenni ehhez a gondolathoz, hogy az érem másik oldalán egy másik – a következőkben tárgyalandó, és egyáltalán nem elhanyagolható – belső motiváció áll, mégpedig a kudarcból való félelem. Ugyanis az lehet, hogy ha minden diákot figyelembe veszünk, akkor relatív, hogy kinek mi a siker. Azt azonban egyértelműen kijelenthetjük, hogy az elégtelen értékelést a diákok túlnyomó többsége kudarcként fogja kezelni.

Az előző gondolatunkat követve elérkeztünk a belső motivációk áttekintésénél a harmadik állomásunkhoz, a már korábban említett kudarcból való félelemhez. Mint azt már láthattuk, ha másban nem is, abban mindenképpen hatékony, hogy a diákok arra törekedjenek, hogy a tanár által támasztott minimális követelményeknek valamilyen szinten megfeleljenek. Ebben az esetben is két oldala van az éremnek. Hisz mint ahogyan azt sem vagyunk hivatottak eldönteni, hogy a diákoknak mi számít sikernek, úgy azt sem mi határozzuk meg, hogy egy adott diáknak mi a kudarc. Ezt könnyen beláthatjuk. Vegyünk egy jó, valamint egy közepes tanulót. A jó tanulónak minden bizonnyal azon tantárgyak esetében, amelyben jó eredményeket ért eddig el egy közepes értékelés már kudarcnak számít, míg a közepes tanuló

mindenképpen sikernek fogja értékelni, s örülni fog. Ebben a megvilágításban elmondhatjuk, hogy minden diák a saját mércéje szerint fogja magát motiválva érezni.

A diákok mások elismerésére való törekvése ritkán irányul diáktársaik felé. Ez leginkább a jobb tanulók esetében fordulhat elő, olyan tanulóknál, akik egymás riválisai. Más esetekben viszont elmondhatjuk, hogy vagy szülői, vagy tanári elismerésre törekednek a diákok. Mind a két esetben lehetnek buktatók, hisz a szülők egy része ugyanúgy van a megszerzett jegyekkel, mint maga a diák. Elfogadja, hogy gyermeke közepes tanuló, és dicséretet, elismerést is ehhez mértén adja. A tanár részéről hasonló, mindannyiunk számára jól ismert problémakörrel van szó, és ez nem más, mint a diákok beskatulyázása. Ma már pszichológusok által bizonyítva van, hogy a tanárok jobban odafigyelnek, meghallgatják a jobb – vagy az általuk jobbnak vélt – tanulókat. S kevésbé – vagyis, hogy pontosabban fejezzem ki másként, és itt ez negatív hatású – figyelnek a gyengébb tanulókra.

Utolsóként még meg kell említenünk a belső motivációk közül az egyik leglényegesebbet, a diákok célkitűzéseit. Ez mindenkinél megvan, és persze mindenkinek más-más. Egy biztos: A diákok egyéni célkitűzéseiről sajnos legtöbb esetben elmondhatjuk, hogy a középiskolában még nem összpontosulnak a tanulásra, és a tanulásból következő dolgok köré.

Most, hogy megvizsgáltuk a belső motivációkat is, továbbléphetünk. Ugyanúgy, ahogyan a külső motivációkban, úgy a belső motivációkban is találhatunk valami közösét. A kérdés tehát: Milyen közös motívumot, motívumokat találunk a belső motivációban? Ha figyelmesen olvastunk, talán hiányosnak tűnhettek az egyes belső motivációk. Esetleg e hiányosság az általunk keresett közös dolog? S ha igen, előrébb jutunk-e a segítségével a középiskolai sikeres motivációhoz? Mindenekelőtt konkrétan rá kell világítani a belső motivációkban lévő közös hiányosságra. Ez pedig nem más, mint a globalitás hiánya a diákok gondolkodásában, szemléletében. Ezalatt azt értem, hogy a legtöbb diák még a középiskola negyedik osztályában sem feltétlenül azért tanul, hogy a tanulmányai alapján majd továbbtanulhasson. A továbbtanulással legtöbbjük csak akkor szembesül, amikor a jelentkezési lapokat ki kell tölteni, majd le kell adni. Ilyenkor ezek a diákok a tanulmányi eredményük, valamint az érdeklődési körük függvényében próbálnak meg megfelelő felsőoktatási intézményt választani. Ilyenkor is olyan helyre próbálnak bejutni, ahol kisebb „versenyre” számítanak. Pedig minderre nem lenne szükség, ha ezek a diákok komolyabban vennék a középiskolai éveket, s már az elején – talán szabad azt mondanom, hogy a középiskola megválasztásánál – célul tűznék ki maguknak, hogy mit szeretnének csinálni az életben, és erre törekednének.

Ehhez viszont megfelelő motivációra, valamint segítségre és támogatásra van szükségük, mind a szülő, mind a tanár részéről.

Persze az előbb említett globalitás hiánya általánosságban jellemző a diákok gondolkodására, szemléletére. Mint általában minden dologban, úgy ebben is vannak kivételek. Nekünk pedig pont a diákok ellenkező táborával kell jobban törődnünk, hisz ők több odafigyelést igényelnek. Abban persze teljes mértékben egyetérthetünk, hogy mind a tudásvágy, a sikerorientáció, a kudarcból való félelem, a mások elismerése, és végül a célkitűzés erősebben befolyásolja nem csak a diákokat, hanem általában az embereket. Mindezek alapján most már bátran kijelenthetjük, hogy a sikeres motiváció a belső motivációk segítségével valósulhat meg. A tanárnak meg kell próbálnia minél inkább pótolni a hiányosságot a diákok szemléletében, segíteni kell nekik céljaik meghatározásában, valamint azok elérésében is.

Célunk tehát megoldások keresése, néhány – régen megfogalmazott és ma is kiválóan alkalmazható – lehetőség felvázolása arra, hogyan tudnánk a meglévő belső motivációt külső ráhatásokkal minél hatékonyabbá tenni. A most következő esetleges megoldásokat a saját tapasztalataim, valamint az ismerőseimmel, barátaimmal folytatott beszélgetések segítségével próbálom meghatározni.

Elsőként talán a tananyag összeállításánál áttekinthetjük, hogy mi minek a megértését, elmélyítését teszi könnyebbé. Ezt a tananyag összeállításánál általában fő szempontként mindenütt figyelembe vesszük. Ne sajnáljuk az időt arra sem, hogy „elmeséljük” – például már az első héten – mennyire hasznos későbbi életpályájukon, esetleg a mindennapi életben a tárgy tananyaga. Ezt illusztrálhatjuk néhány konkrét példával is. Megmutathatjuk azt is, hogy hogyan alkalmazhatják majd az órán, ill. a középiskolában szerzett tudásukat, de azt is, hogy milyen nagy hibát követhet el az akaratán kívül az, akinek nem volt lehetősége ilyesmit tanulni. Tudatosítani kell, hogy felnőtt életükben nélkülözhetetlen szükségük lesz a tanultakra. A tanulóknak éreznie kell, hogy az elsajátítandó ismeretanyag számukra fontos, és szükséges céljaik eléréséhez.

Mindehhez hasznos lehet részcélok megjelölése. Ehhez például több rövid számonkérést is alkalmazhatunk. Ez lehet, hogy részünkről többletmunkát, de a tanulók részére könnyítést, személyes örömet jelenthet. Valamint ne feledjük, hogy a pozitív hozzáállás az anyag elsajátításában mélyíti a tudást. Összességében elmondhatjuk, hogy arra kell törekednünk,

hogyan az oktatás folyamata, és annak egyes részei folyamatosan és állandóan, új és új érdeklődéseket szülnének.

A tanulók munkáját mindemellett folyamatosan jutalmazhatjuk. Például „piros pont”-tal való rendszeres értékelésről az általam megkérdezett tanulók véleménye a következő volt: Bár néhányan egy picit gyerekesnek tartották, de mégis úgy gondolták, hogy ez jó móka is egyben. Éppen ezért a többség nemcsak jó ötletnek tartja, de szereti is.

Sokat segíthet az is, ha kérdéseik feltevésére bátorítjuk a tanulókat. Ez azért is jó, mert segíthet bennünket a tananyag későbbi átdolgozásában és az évfolyamot, tanulóink képességeit is világosabban látjuk. A megkérdezettek közül többen megjegyezték, hogy az ilyen és ehhez hasonló órákon számukra ezt érezték a legjobbnak, és nem utolsó sorban leghasznosabbnak.

Segítségünkre lehet a jó légkör, derűs hangulat. Saját tapasztalataim, és az ismerőseim, barátaim által adott válaszok is azt mutatják, hogy ha a tanulók látják, hogy jobb eredményük az oktatónak is cél, örömet okoz és fontos, akkor az is motiválja őket. Így több visszajelzést kaphatunk, valamint könnyebb megbeszélni a nehézségeket is.

Lelkesedésünk, valamint az anyag szépségeinek, érdekességeinek kiemelése is nagymértékben felkeltheti a tanulók érdeklődését. Ezáltal ugyanis kialakíthatunk egy úgynevezett ösztönös kíváncsiságot, mely esetünkben a tanulás egyik legfőbb mozgatója lehet. Ha jobban belegondolunk, ez lehet majdhogynem az egyetlen eszközünk, amellyel az oktatás eredményét tartóssá tehetjük. Érdemes erre időt, energiát szánni még akkor is, ha az a szokott anyagnál nehezebb. Minél több érdekes példát mutassunk be. Fogalmazzuk át a feladatokat úgy, hogy már a szövegére is felfigyeljenek, hogy emlékezetükbe véshessük. Készülhetünk úgy, hogy amikor észrevesszük a fáradtságot, a figyelem lazulását, akkor „elővehessünk” egy olyan történetet, példát, amire mindenki felfigyel, ami ébresztőként hat.

Az oktatásban rugalmasnak kell lennünk. Hallgassuk meg a tanulók ötleteit, véleményeit, észrevételeit, kapcsoljuk be a tanulókat a tananyag és a feladatok aktív feldolgozásába, hiszen ha belegondolunk, a diák tanul, s nem az oktató. S ne feledjük, az eredményesen megoldott feladatnak is – nem csekély – motiváló értéke van.

Most, hogy megnéztük mi is az a motiváció, s megvizsgáltuk annak ideglélektani hatásait, beláthatjuk, hogy nincs olyan pillanat az ember életében, amikor ne lenne motiválva. Ezután a motivációkat két csoportra osztottuk, majd mind a külső, mind a belső motivációkat nagyító alá vettük, és részletesen megvizsgáltuk.

Mindezen vizsgálódásainkból megtudtuk, hogy a külső motivációk nem igazán alkalmasak arra, hogy a tanár eszközként használja a diák motiválására, ellentétben a diákok belső motivációival. Hiszen mint azt vizsgálatunk is mutatta, a belső motivációk nagyobb mértékben hatnak a diákok cselekedeteire, sokkal inkább ösztönzik őket, s nem utolsó sorban a tanárok is jobban tudják azt bizonyos módszerekkel befolyásolni. A továbbiakban megvizsgáltunk néhányat e módszerekből. Remélem, hogy ezen a módszerek vizsgálata eléggé körvonalazta a sikeres motiváció megtalálásának módját, mikéntjét, azért is, mert úgy gondolom, hogy ennél közelebb nem is nagyon lehetne megközelíteni magát a sikeres motivációt. Hisz hiába az érintetek vallottak, mégis az eredmények, következtetések meghatározásánál a többség véleménye volt a döntő. Azaz vannak, s mindig is lesznek diákok, akiknek más a véleménye, felfogása a dolgokról, s ezáltal másfajta, egyéni motiválást kell náluk alkalmazni.

Eszközök az informatikában

Míg más órákon könyvek és füzetek állnak a diákok és a tanár rendelkezésére, addig az informatikában ezek az eszközök eléggé elhanyagolhatóak, ugyanis más, hatékonyabb eszközök helyettesítik ezeket. Természetesen vannak nagyon jól használható könyvek, és a diákok jegyzetelésének a fontosságát sem lehet vitatni, de ettől még nyugodtan elmondhatjuk, hogy informatika órán más eszközön van a hangsúly. Ilyen elmaradhatatlan eszköz az órán maga a számítógép.

Önmagában a gépteremben lévő számítógépek közül egy sem ér túlságosan sokat a tanítás szempontjából. Az csak egy hardver. Az, ami ténylegesen hatékony eszközzé teszi, ami nélkülözhetetlen róla az a szoftver. Bár a tananyag része a számítógépek hardver felépítése is, ennek ellenére a hangsúly a számítógépek gyakorlati alkalmazásán van. Vagyis olyan feladatok elvégzésén melyekhez nélkülözhetetlen maga a számítógép és a rajta lévő megfelelő szoftver. A tanítási órákon e szoftverek használatát oktatjuk, valamint azt, hogy egy-egy specifikus feladatot mely program segítségével, és hogyan tudjuk megoldani. Ilyen szoftverek például a szövegszerkesztők, táblázatkezelők, adatbázis kezelők, prezentációkészítők, képnézegetők, képszerkesztők, de nem utolsó sorban maga az operációs rendszer is.

Viszont egy számítógép, mely rendelkezik minden fontos programmal, melynek használatát meg akarjuk tanítani még nem elég. Fontos, hogy a hatékony tanulás érdekében minden diák külön gépen tudja követni az óra menetét, ne kelljen megosztania a figyelmét, s ezáltal gyorsabban haladhasson az óra anyagának feldolgozásában. Az már bizonyított, hogy olyan órákon, melyek a gyakorlatra helyezik a hangsúlyt sokkal hatékonyabb, és praktikusabb, ha a diákok létszáma limitált, tehát nem mint egész osztályos foglalkozásban tanítjuk őket, hanem csoportokra osztva. Az osztályok ilyen csoportokra való osztása a tanár dolgát is megkönnyíti, mert jobban tud figyelni a diákokra, ezáltal hatékonyabban tudja őket segíteni.

Természetesen a tanteremben rendelkezésünkre álló gépeknél alapkövetelmény, hogy hálózatba legyenek kötve, valamint az, hogy mindegyik rendelkezzen interneteléréssel. Hiszen ezek nélkül a feltételek nélkül nem igazán beszélhetünk weboktatásról.

Mint minden más tantárgynál, ugyan úgy az informatikában is nagyon hasznos, és nélkülözhetetlen eszköz a tábla. A diákok a tanár által táblára írt dolgokat írják a füzetbe – sokszor szóról szóra. Természetesen az még jobb, ha a füzetbe bekerülnek a táblán lévő

dolgok mellett a tanár szóbeli magyarázatai az egyes dolgokhoz. Sokszor azonban a tantárgy jellegéből adódóan – órán nem füzetbe dolgozunk, hanem számítógépekkel – a tábla kevésnek bizonyul a tanár számára. Nem lehet pontosan azt visszaadni a táblán, amit éppen a diákoknak a monitoron látniuk kellene. Pedig sokszor kerülhetünk olyan helyzetbe, hogy az általunk elmondott, tanított anyagot legjobban úgy tudjuk szemléltetni, bemutatni az osztálynak, ha azt egy konkrét példán keresztül mutassuk meg, minden részletével. Régen ilyen esetekben a tanár a diákokat egy géphez hívta, hogy mindenki megfigyelhesse az adott példát, hogy ezután mindenki a saját gépén oldja meg a feladatot a látottak segítségével. Ez sok időkieséssel jár, és sokszor nem is biztos, hogy ezután egy adott feladatot a gyerekek sikeresen tudtak megoldani, mert elfelejtettek egy-két fontos lépést, ami azt eredményezte, hogy vagy megkérdezték újra a tanárt, vagy rosszabb esetben mellettük ülő társukat, akit ezzel zavartak a feladat megoldásban. Ennek a problémának a megoldására kellett egy olyan eszköz, mely segítségével úgy mutathatjuk meg gyakorlatban az adott program használatát, egy adott feladat megoldásának lépéseit, hogy a diákok akár párhuzamosan tudjanak haladni a tanárral. Ez az eszköz a ma már az oktatásban elterjedtnek mondható projektor. A fenti probléma megoldásán kívül még nagyon sok olyan terület van, ahol javíthatjuk vele az oktatás hatékonyságát, és mindemellett megkönnyíthetjük a saját dolgunkat. Gondoljunk csak egy adott tananyag leadásánál egy jól elkészített prezentációra, ami persze tartalmazhatja azokat az információkat is, amiket a táblára íránk, de lényegében az a fő célunk vele, hogy megfelelően szemléltesse azokat a dolgokat, amikről beszélünk, hogy a gyerekek jobban megérthessék azt. A projektor ilyen hatékony felhasználásánál a tábla, mint segédeszköz jelenik, melyet esetlegesen arra használhatunk, hogy problémásabb anyagrészeket magyarázzunk el más szemszögből.

A számítógép, projektor kettőst kiegészítve egy harmadik eszközzel, az aktív táblával elmondhatjuk, hogy e három eszköz a mai oktatás legmodernebb és leghatékonyabb kombinációja.

Az aktív tábla egy érintés-érzékeny felület, melyre a számítógép monitorának képét egy projektor segítségével kivetíthetjük, azon keresztül a számítógépet vezérelhetjük, az azon megtalálható valamennyi alkalmazást elindíthatjuk, futtathatjuk. Ha a számítógéphez hangszórót, DVD- vagy videólejátszót csatlakoztatunk, akkor multimédiás megoldásokkal is élhetünk. Ha a számítógépet csatlakoztatjuk az Internetre, az aktív tábláról közvetlenül hozzáférhetünk a kívánt weboldalakhoz.

A hagyományos tábla esetén a táblát gyakran töröljük, ilyenkor annak tartalma elvész, illetve annyi marad meg belőle, amennyit a tanulók a füzetükben rögzítettek. Ezzel szemben a digitális tábla táblaképeit elmenthetjük, ezeket az óra későbbi szakaszában, vagy bármelyik másik órán elővehetjük, és a képi megjelenés segítségével az elhangzottakat felidézhetjük.

Az önálló tanulás folyamatát segítheti, hogy a tanóra vázlatát, a táblaképeket tartalmazó fájlt kinyomtatva a gyermek kezébe adhatjuk, vagy az iskolai hálózaton elhelyezhetjük tanulóink ennek segítségével készülhetnek a következő órákra.

A teljes óra anyagát esetleg egy részét előre rögzíthetjük könnyítve ezzel munkánkat. Egy esetleges helyettesítés esetén a helyettesítő tanár egyszerűen levetíti a diákoknak az általunk rögzített információkat. Természetesen az előzetes rögzítés ugyanúgy időbe kerül.

A hagyományos falitábla használata a tanórákon – még ha előre tervezett is – nagyon sok esetben válik rendezetlenné. Az aktívtábla egyik fontos előnye, hogy táblaképeink nem csak tervezhetők, de előre el is készíthetők, így tudományterületspecifikus hasznosságán túl az esztétikai nevelés igényeit is kielégíthetjük. Előre elkészített tananyagainkat megfelelő szoftverek segítségével interaktívvá tehetjük, így diákjaink a passzív befogadóból aktív közreműködőkké válnak, ezzel is segíthetjük a megértés, a tanulás folyamatát.

A fentiekben fel lettek sorolva azok az eszközök, melyeket alkalmazva az informatika órán hatékonyabbá tehetjük a tanítást. Van azonban még egy igen fontos eszköz, amit nem hagyhatunk számításán kívül. Különösképpen azért nem, mert az informatikai világban ez fejlődik talán a legnagyobb mértékben. Ez az eszköz nem más, mint az internet.

Ma már közhelyszámba megy, hogy az élet minden területén milyen forradalmi változásokat hozott az internet. Az egyik ilyen egyértelműen érzékelhető (és folyamatos) változás a szoftverek használatában, a felhasználók hozzáállásában és a szoftveriparban megy végbe. Időszámításunkat „internet előtti” és „internet utáni” korszakokra osztva az alábbiakat figyelhetjük meg.

A nyolcvanas évektől kezdve – a PC-k elterjedésével – természetessé vált, hogy a szoftvert termékként kezelik. Boltban árulják, dobozba csomagolva, egyszerű árucikként. Végül is a szoftver – bár nem kézzelfogható dolog – ugyanolyan eszközzé vált, mint egyéb használati tárgyaink. Ez a szemlélet jellemző a számítógéppel segített oktatás legismertebb formájára, a CD-ROM-on terjesztett multimédiás oktatóprogramokra is. Itt az oktatás hagyományos médiumainak kiterjesztésén és az önálló tanuláson van a hangsúly. A CD-ROM-on az oktatási

anyagok és a megjelenítést végző szoftver általában zárt rendszert alkotnak. Egy tárgyról van szó, amelyet ugyanúgy emelünk le a polcra, ha használni akarjuk, mint egy könyvet.

Az „internetkorszakban” egyre inkább világossá válik, hogy az internet az oktatás elválaszthatatlan, szerves része. Egyrészt világméretű információs bázis, azt hiszem bátran kijelenthetem, hogy bármely általános iskolai, vagy egyetemi tantárgyhoz találunk a „net-en” valamiféle oktatási segédletet. Az informatika az egyik leggyorsabban fejlődő iparág, ezáltal a legfrissebb hardver és szoftver információkhoz legegyszerűbben juthatunk hozzá, ha leülünk egy számítógép elé, amely valamilyen internet kapcsolattal rendelkezik. Véleményem szerint ma egyetlen tanár sem nélkülözheti az internet adta lehetőségeket

Az internet elengedhetetlen oktatási segédletté vált. Természetesen a tanár feladata, hogy e hatalmas információs bázisból mi az, amit a tanórákon hasznosít, de e lehetőségtől való teljes elzárkózás súlyos hiba. Annál is inkább vallom ezt a nézetet, mivel dolgozatomban témájának is szerves része, hogyan lehet megismertetni a diákokkal az internet adta lehetőséget. Hogyan tudnak hatékonyan információhoz jutni az interneten, olyan információkhoz, mely akár saját célú, akár más óra tananyagához tartozik. A megtalált információkat hogyan tudják leghatékonyabban szűrni, és a már kiszűrt információkat hogyan tudják legjobban hasznosítani. Ez persze egy lehetősége az internet használatának. Azt is meg kell tanulniuk, hogyan tudnak maguk is megfelelő módon – figyelve formai, etikai szabályokra – információt elhelyezni az interneten. Nem utolsó sorban pedig, hogyan tudnak élni az internet adta kommunikációs lehetőségekkel.

Weboktatás területei

Mint azt már dolgozatomban írtam, a weboktatásnak több területe van, amit a tanárnak meg kell tanítani a diákokkal. Mivel órán is hálózatba kapcsolt számítógépekkel dolgozunk, melyek – jó esetben – internethozzáféréssel is rendelkeznek, ezért elengedhetetlennek tartom magának a hálózat fogalmának és felépítésének minimális szintű megismertetését. Ilyen ismeretekre ma már bárkinek szüksége lehet, gondoljunk csak a legtöbb családnál megtalálható szereld magad ADSL csomagokra, vagy akár a kábel internetre.

A többi területről is eset már szó dolgozatomban. Ilyen terület az információ keresése az interneten. Itt valójában nem egyszerű információkeresésről van szó, hanem céltudatos, hatékony információkeresésről, ami nehéz, hisz fontos, hogy milyen információt milyen keresővel, milyen kulcsszóval, esetleg kulcs szavak kombinációjával kell keresni. Itt azonban nem ér véget a dolog, hisz ma már az internetes oldalak olyan nagy mennyiségű információval bombázzák az embereket, amiket sokszor nagyon nehéz kezelni. Ezért is van, hogy a sikeres keresés nem ér véget e töménytelen információ megjelenítésével. Ezeket az információkat szűrni kell. Meg kell tanítani a diákoknak, hogyan tudják leghatékonyabb kiszűrni a számukra fontos információkat. Ez persze itt még inkább mennyiségi szűrés. Fontos, hogy tudatosítsuk velük a minőségi szűrés fontosságát is.

Következő fontos terület az információ elhelyezése, hogy az mindenki számára elérhető legyen. Ebbe a területbe természetesen beletartozik magának az információnak az elkészítése, mind formailag, esztétikailag, mind tartalmilag, utolsó lépésként az elkészített saját információ interneten való megosztása, elhelyezése.

A területeknél még meg kell említeni az interneten különböző módon történő kommunikációs formákat, azok használatát. Itt nagyon fontos megismertetni a tanulókkal azon szabályokat melyeket be kell tartani a különböző kommunikációs formáknál, és azon veszélyeket, amiket ezek magukba foglalnak.

Végül fontos, hogy ha minimálisan is, de szót ejtsünk az internet adta inkább kényelmi lehetőségekről, mint például az on-line vásárlás, on-line bankszámla kezelések, stb. Ezek bár megkönnyíthetik életünket, ennek ellenére nem szabad sosem elfelejteni milyen veszélyeket rejtene, és ezért használatuk is nagyobb odafigyelést igényel.

Alapvető hálózati ismeretek

Mielőtt belekezdenénk a weboktatás fő területeinek tanításába, fontos néhány alapvető dolgot megtanítsunk a hálózatokról, azok felépítésükről. Nem kell túlságosan részletesen taglalni e részeket, hanem éppen csak annyira, hogy a fő alapgokkal tisztában legyenek a tanulók. Az egyes fogalmaknál, mint például a hálózat fogalma is, törekedjünk az egyszerűsége, de mindemellett a pontosságra is. Amit mondunk az nem feltétlenül fontos, hogy teljes legyen, de korrektnek és egésznek kell lennie.

Amire kitérhetünk órán – természetesen csak bizonyos részletességgel –, a hálózat, mint fogalom, kialakulásának története, előnyei, hátrányai. Beszéljünk a hálózatok fizikai megvalósításáról, a hálózatok csoportosításáról, fizikai kiépítésben előfordulható átviteli közegekről, hardverelemekről, topológiákról.

Ezek után térjünk ki, hogyan zajlik a hálózatokon az egyes gépek kommunikációja, milyen protokollok vannak. Mutassuk meg nekik, hogyan tudjuk beállítani a hálózatot a számítógépen. Itt néhány fogalmat próbáljunk minél egyszerűbben, érthetőbben elmagyarázni. Ilyen fogalmak többek között az IP, DNS és az átjáró. Ezek után rátérhetünk az internetre, és az azon való keresésre.

Információ keresése az interneten

Amikor valamilyen információra van szükségünk, leggyakrabban az interneten keressük azt. Fontos, hogy az általunk talált információ a legteljesebben lefedje, minél pontosabb választ adjon a keresett információra, hiszen e dolgok arányában mérhetjük egy keresés sikerességét. Az emberek nagyon sok mindenre kereshetnek az interneten. Hobbijukkal, munkájukkal kapcsolatosan. Fontos, hogy a diákok ne csak játékot, szórakozást lássanak az internetben, hanem használati eszközt, mely a saját dolgukat is megkönnyíti, valamint amivel ki tudja egészíteni az iskolában tanultakat azon információkkal, amik a világhálón megtalálhatóak.

Az interneten elérhető információ mennyisége olyan hatalmas, hogy nem is igazán tudjuk mérni. Egy keresés során ebből a hatalmas információhalmazból kell kiválasztanunk, megtalálnunk a nekünk megfelelőt, ami nem is olyan könnyű sok esetben. Ezért kell a diákokat is megtanítani ezen a hatalmas információ sztrádán való eligazodásra. Hogyan keressen adott dologra, hogyan szűrje ki a találatok közül a neki megfelelőt a sok érdektelen,

zavaró információ közül. Természetesen ma már a kereső oldalak olyan fejlett algoritmusokat használnak amivel, leggyakrabban az általunk keresett oldalakat adják eredményül, de azért vannak esetek, amikor tovább kell lépni, és használni kell ezen oldalak speciális lehetőségeit. Éppen ezért ezeket is mutassuk meg a tanulóknak. Mutassuk meg nekik például, milyen speciális keresési beállításokkal szolgál az egyik legismertebb keresőoldala a Google.

Amint az láthatjuk egy áttekinthető, esztétikus oldalt kapunk, melynél továbbra is kulcsszavak segítségével tudunk megadni egy keresést. Annyiban különbözik a normál keresésnél, hogy adott kulcsszavakat különböző feltételek szerint tudunk csoportosítani. Például, hogy tartalmazza e a talált oldal a megadott kulcsszavakat vagy sem. Mindemellett lehetőségünk van keresni többek között adott nyelv, régió, dátum, fájlformátum alapján.

Információ elhelyezése

Mint ahogy azt már taglaltam, az információ keresése mellett az információ mindenki számára elérhetővé tétele, vagyis interneten való elhelyezése szintén nagyon fontos weboktatási témakör, amit a diákoknak szintén fontos sajátítaniuk.

Szerencsésnek mondhatom magam, mert fél évig lehetőségem nyílt informatikát oktatni volt iskolában, a nyíregyházi Széchenyi István Közgazdasági, Informatikai Szakközépiskolában. Az általam tanított osztályok között volt 9, 10 és 11-dikes, ezáltal nagyobb rálátást kaphattam arról, hogy az egyes évfolyamok mennyire voltak fogékonyak a tananyag ezen részére, mennyire okozott nehézséget nekik ez az anyagrész.

Dolgozatomban az információ elhelyezése alatt magának a HTML nyelvnek az alap szintű elsajátítását, valamint az elkészített dokumentumok interneten való elhelyezését értem. A HTML nyelv tanítása során a fapados – jegyzettömböt, valamint a formázó utasításokat, tagokat használó - módszert alkalmaztam, hogy a diákok nagyobb rálátást kapjanak a nyelvről. A 11-dikes osztályok már előzőleg tanultak HTML szerkesztést a Mozilla Firefox internet böngésző – WYSIWYG – szerkesztőjének segítségével, ennek ellenére úgy láttam, ebből semmilyen előnyük nem származott a többi osztállyal szemben. Jómagam azon az állásponton vagyok, hogy a diákoknak először magát a HTML nyelv felépítését, működését, logikáját, valamint néhány alap formázóutasítást kell elsajátítani, majd ezután megismertetni velük a HTML dokumentum készítésének egyéb – különböző szerkesztők által kínált – lehetőségekre. Ez persze nem jelenti azt, hogy nem használhatjuk ki e szerkesztők bizonyos funkcióit, a HTML nyelv tanítása során, melyek nem csak megkönnyítik a munkánkat, de a diákoknak is kényelmesebb környezetet teremtenek a tanultak megértéséhez. Ilyen funkció például a szöveg szintaktikai kiemelése, esetleg automatikus kiegészítése, melyre az egyszerű szövegszerkesztők, mint például a jegyzettömb – amivel persze tökéletesen lehet HTML dokumentumot készíteni ettől függetlenül – nem képes. Ez persze nem is fontos. Itt ugyanis az számít, hogy a diákok tisztában legyenek a HTML és az egyszerű TXT szövegfájl közötti kapcsolatra.

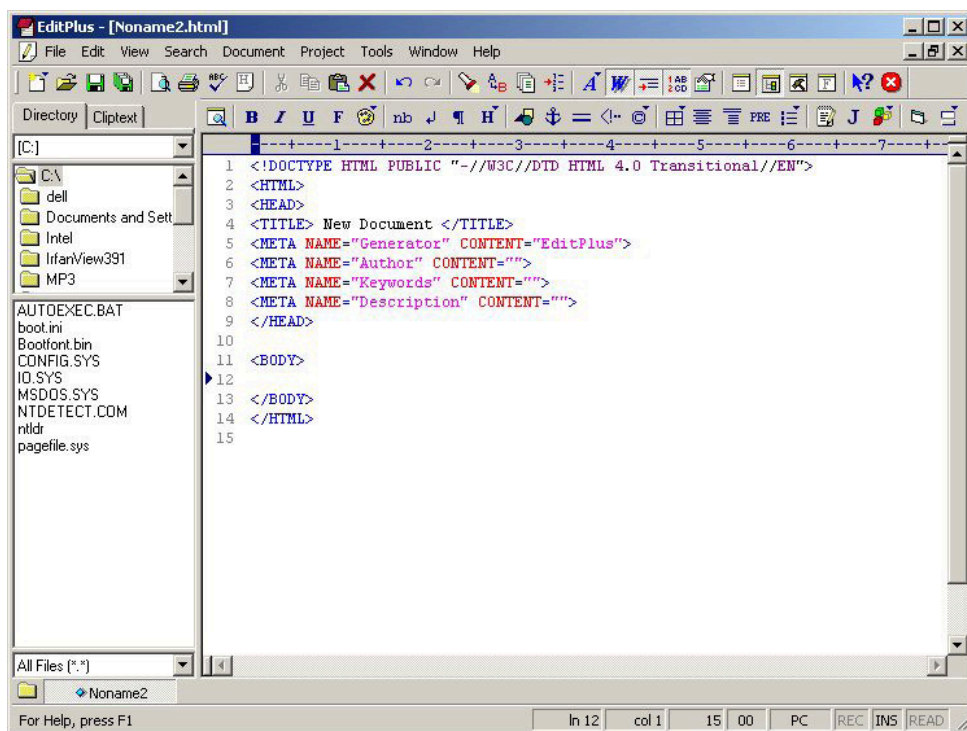
Annál is inkább, mert a HTML nyelv tanításának már a legelején előjöhethet olyan kérdés, hogy: „Ezt lehet Word-ben is?” A választ természetesen az, hogy igen, mégis azért ezzel a kérdéskörrel óvatosan kell bánni. Tudatosan kell szoktatni a diákokat arra, hogy minden munkához, amit számítógéppel végeznek a lehető legmegfelelőbb, legkényelmesebb programokat használják. Az előbb említett kérdésnél maradva, hamar beláthatjuk, hogy a Word nem pont az, amire nekünk szükségünk van, ugyanis funkcióját tekintve nem egyszerű formázatlan szövegfájl készítésére tervezték.

Mivel dolgozatomnak nem célja felsorolni a különböző szerkesztőket, valamint azok előnyét hátrányát, ezért egy általam preferált szerkesztőre szorítkoznék, azt is csak nagyvonalakban ismertetném. Ez a szerkesztő az EditPlus 2.

EditPlus 2

Sok indok szól az EditPlus 2 mellett. Többek között az, hogy ez a szerkesztő nem csak a Windows beépített szöveges fájlok létrehozására szánt Jegyzettömböt váltja le, hanem

esetlegesen egyes programozási nyelvekhez szánt fejlesztői környezeteket is. Ugyanis ez a szerkesztő nemcsak a HTML dokumentumokat emeli ki szintaktikailag, hanem különböző programozási nyelvekhez is – mint például a C, Pascal, Java – le lehet tölteni a szintaktikájuk kiemeléséhez szükséges kiegészítést.



Új dokumentum
nyitásakor
választhatunk,
hogy normál
szöveget
kívánunk e
létrehozni, vagy
esetleg a már
kiegészítésként
letöltött
támogatott
dokumentum
formátumokat,
melyek esetében

a szerkesztő felépíti annak alap struktúráját. Ez látható a fenti képen is. Mivel HTML dokumentumainkat szövegfájlként készítjük el, és a formázás során az egyes tagokat nekünk kell kézzel begépelni, igen hasznos lehet a szerkesztő azon tulajdonsága, mellyel, mint egy átmenetet képez a szöveges, kézzel való szerkesztés és a WYSIWYG szerkesztők között. Ugyanis az eszköztáron megtalálhatók azon gombok, mellyel például szöveget tudunk félkövérré, dőltté alakítani, vagy táblázatot, képet szúrhatunk be. A különbség, hogy e funkciók tisztán szöveges módban működnek, és használatukkal az adott tagokat rakja a dokumentumba a szerkesztő, ezzel is megkönnyítve a munkánkat. Természetesen az egyes tagok paramétereit nekünk kell megadni. Az átmenet ott figyelhető meg, hogy az EditPlus 2 tartalmaz beépített böngészőt, melynek segítségével a kézzel, valamint a szerkesztő által nyújtott kényelmi funkciók használatával részben, egészben elkészített HTML dokumentumunkat meg tudjuk tekinteni úgy, mintha böngésző nyitotta volna meg, ezzel ellenőrizhetjük magunkat folyamatosan a szerkesztés alatt. Ebben a böngésző módban viszont

nincs lehetőségünk a dokumentum szerkesztésére, ahhoz újra szerkesztő módra kell váltanunk. Sajnálatos módon ez a szerkesztő nem ingyenes.

Alternatívaként használhatjuk esetleg még a Notepad++ forráskód-szerkesztő mely az. Ez nem rendelkezik az EditPlus 2 minden említett tulajdonságával, de ettől függetlenül szintén hatékony eszköz, mivel ugyan úgy képes a számos programozási nyelv szintaktikai kiemelésére.

A HTML nyelv alapjai

Miután megvizsgáltuk lehetőségeinket a megfelelő szerkesztő kiválasztásánál, és sikerült is találnunk számunkra ideálist, következhet magának a HTML nyelvnek az elsajátítása. Ebbe természetesen beletartozik annak kialakulása, minimális történeti áttekintése is.

A nyelv elsajátításának első lépésként tisztázni kell a diákokkal mi is az a HTML nyelv, valamint, hogy mikből, hogyan épül fel. Fontos kitérnünk a platformfüggetlenség fogalmára – hisz a HTML nyelvet is nagymértékben annak mondhatjuk –, mellyel talán a diákok először találkoznak. Az egyik legfontosabb fogalom itt a HTML tag, formázóutasítás, ami egészen addig, amíg egy egyszerű mégis sokatmondó és látványos példát nem látnak, addig a levegőben lóg. Éppen ezért a tag, mint fogalom, valamint a HTML strukturális felépítésének a magyarázatát össze lehet vonni, hiszen magát a struktúrát is tagok építik fel. Erre az alábbi példa megfelelő:

```
<HTML>
<HEAD>
<TITLE>A címe, ami a böngésző címsorban jelenik meg!</TITLE>
</HEAD>
<BODY>
A dokumentum helye.
</BODY>
</HTML>
```

Jó, ha felhívjuk a diákok figyelmét, hogy miközben szerkesztik a dokumentumukat, minden lényegi változtatás után mentsek munkájukat, és folyamatosan ellenőrizzék magukat böngészőben. Hívjuk fel a figyelmüket arra, hogy HTML dokumentumok nevénel szóközt és ékezetet ne használjanak, utalva a hivatkozásokra és azt ott felmerülő problémákra.

Ennek a példának a segítségével bemutathatjuk a HTML dokumentum felépítését, struktúráját, valamint a tagok működését. Itt azért fontos amennyire csak lehet általánosan is

megfogalmazzunk, hogy egy tag, mint utasítás mire fejt ki a hatását. Valamint, mivel a továbbiakban célunk a különböző formázóutasítások elsajátítása, hasznos lehet, ha a diákok a füzetükbe is rögzítik a tanultakat. Ugyanis a füzet, és az órai munka során létrehozott dokumentumok együtt már elegendőek a diákok önálló, otthoni tanulására, ismétlésére. Ezzel lehetőségünk nyílik arra is, hogy a tanultakat a későbbiek során, írásban is számon kérhetjük.

A füzetbe kerülhetnek az általánosan megfogalmazott szabályok, valamint az egyes tagok a paraméterükkel. Természetesen ügyelnünk kell, hogy a füzetbe került dolgok átláthatóak legyenek, ezért is jó, ha bizonyos dolgokat megpróbálunk megfelelő magyarázat mellett általánosítani. Munkánk elején ilyen általánosításra van lehetőségünk a tagok esetében is:

```
<TAGNÉV PARAMÉTER1="ÉRTÉK1" PARAMÉTER2="ÉRTÉK2" PARAMÉTER3>  
SZÖVEG, amelyre a formázóutasítás vonatkozik  
</TAGNÉV>
```

A fenti példán egy tag általános alakját láthatjuk, amit a diákok rögzíthetnek a füzetbe. Fontos, hogy kihangsúlyozzuk a nyitó és záró tagokra vonatkozó szabályt, azaz, hogy a nyitó tag esetén a tag neve megegyezik a záró párjának nevével a megelőző / jel kivételével. Ugyancsak nagyon fontos a paraméterek megadásának módja. Paraméterek a nyitótagban, a tag neve után egymástól szóközzel elválasztva szerepelnek. Vannak önálló magát képviselő paraméterek, valamint olyanok, amik értéket várnak. Ebben az esetben az értéket egyenlőségjel segítségével, idézőjelek között kell megadni. Érték lehet szám például betűméretnél, vagy szöveg betűtípusnál. A fenti általánosítás azért is szükségszerű, mert ezáltal a táblára – és természetesen a füzetekbe is – az alábbi módon rögzíthetjük az egyes tagok, paramétereit, valamint ha szükséges annak értékeit:

```
<FONT> - color={ "színnév"  
          "színkód"  
          - face="betűtípus"  
          - size="szám"
```

A tanulók füzete ezáltal sokkal áttekinthetőbb lesz. Persze ez a példa sem teljes, ugyanis egy ilyen táblakép mellé fontos, hogy a füzetbe bekerüljenek a tagok és azok paramétereinek kiegészítő magyarázatai. Ilyen jelen példánál, hogy az adott tagnak van záró párja, valamint, hogy a `size` értéke 1-től 7-ig terjed. Hiába adunk meg azonban általánosítást a tagokra, ha a fenti módon írjuk azokat a táblára nagyon fontos, hogy kezdetben minden óra elején felírjuk

az általános formát, ugyanis amíg a tanulók nem szereznek egy minimális rutint a nyelvből – amit persze csak gyakorlással lehet – addig elvétve előfordulhat, hogy a paramétereket

```
<FONT> - color="blue"
```

formában adják meg. Természetesen vannak más típushibák is, amit a tanulók a HTML dokumentum szerkesztése alatt elkövetnek, de ezekre egy külön fejezetben térnek ki. Mindenesetre itt a kezdeteknél nyomatékosítani kell, hogy a táblára csak a könnyebb áttekinthetőség érdekében kerülnek a tagok olyan formában, a dokumentum szerkesztésekor a tag általános formáját kell használni. Ezt az első példánkon keresztül meg is tudjuk nekik mutatni. Módosítsuk a kódot az alábbi módon:

```
<HTML>
<HEAD>
<TITLE>Virág Róbert honlapja</TITLE>
</HEAD>
<BODY bgcolor="green" text="#0000FF">
Virág Róbert
Születési idő: 1983.07.20.
Hobby: olvasás, túrázás, filmnézés
</BODY>
</HTML>
```

Mivel a példába a tanulók saját nevüket, születési idejüket, hobbijukat, stb. írhatják, ezáltal az picit személyesebbé válik. Azzal, hogy a <BODY> esetében paramétert vezettünk be, rendes példán keresztül is láthatják a tagok és paramétereik használatát, és nem utolsó sorban azáltal, hogy megváltoztattuk a háttér és szöveg színét sokkal látványosabbá is tettük az órai munkát. Sok tanulónak erre a példára már beindul a fantáziája, és érdeklődni kezd az egyes szövegek külön színezésének módjáról, ami nagyon jó, hisz az ilyen tanulók esetében kétséget kizáróan sikerült érdekessé tenni a tananyagot. Van azonban itt még egy dolog, amit szintén körül kell járni. Mivel a <BODY>-ba bekerült a bgcolor paraméter tisztázni kell a tanulókkal mi is az a színkód, milyen alapszínekből épül fel, és azokat hogyan milyen sorrendben kell megadni. Azért is nagyon fontos ez, mert a későbbiekben több formázóutasításnál előjön, így később utalhatunk a már ismert tag paraméterének megadásának azonos analógiájára.

Az eddig elmondottak mellett azért is nagyon hatékony ez a kis példa, mert rögtön megmutat egy másik nagyon fontos dolgot a HTML dokumentumokról, amit véleményem

szerint jó a legelején tisztázni a tanulókkal. A böngésző a HTML kódban minden újsort, valamint egynél több szóközt figyelmen kívül hagy. Ez a példánk esetében úgy jelenik meg, hogy a név mellett rögtön megjelenik a születési idő és a hobbi, hiába kezdték azokat a tanulók új sorokba. Módosítsuk ismételten példánkat azáltal, hogy megtaníttunk egy már-már valódi, nem csak a HTML struktúráját felépítő, hanem annak megjelenését befolyásoló tagot a
-t. Mivel ez a tag amolyan kivétel a többi között, mert nincs záró párja, érdemes a címsorok kialakítását segítő <H1></H1> tagot is becsempészni a kódba:

```
<HTML>
<HEAD>
<TITLE>Virág Róbert honlapja</TITLE>
</HEAD>
<BODY bgcolor="green" text="#0000FF">
<H1>Virág Róbert</H1>
Születési idő: 1983.07.20.<BR>
Hobbi: olvasás, túrázás, filmnézés
</BODY>
</HTML>
```

Ez a példa amennyire egyszerű, olyannyira sokatmondó és hasznos. Megnézhettük vele a HTML dokumentum felépítését, struktúráját, példát láhattunk záró párral nem rendelkező tagra is, valamint több paraméter egyidejű megadására. A paraméter felvett értékéből adódóan megnézhettük a későbbiekben többször előforduló színmegadást színkód segítségével. Azt hihetnénk, hogy többre nem is jó ez a példa, pedig van még egy igencsak hasznos dolog, amit magában hordozhat. Segítségével megtaníthatjuk a diákokat már a munkánk legelején az esztétikus átlátható dokumentum készítésére. Az egyes tagokat itt direkt írtam nagybetűvel – bár a nyelv ezt nem követeli meg – így sokkal áttekinthetőbb a dokumentum, és jobban elkülönülnek a formázóutasítások a formázandó szövegtől. Természetesen, ha nem jegyzetömböt használunk, hanem olyan szerkesztőt mely szintaktikailag kiemeli az egyes tagokat akkor erre nincs szükségünk. Azonban van még egy módszer, mellyel áttekinthetőbbé tehetjük munkánkat. Úgy, hogy figyelve a dokumentum struktúrájára, felépítésére, az egyes tagokat beljebb kezdjük:

```
<HTML>
  <HEAD>
```

```
<TITLE>Virág Róbert honlapja</TITLE>
</HEAD>
<BODY bgcolor="green" text="#0000FF">
  <H1>Virág Róbert</H1>
  Születési idő: 1983.07.20.<BR>
  Hobbai: olvasás, túrázás, filmnézés
</BODY>
</HTML>
```

A HTML tagok

Miután megismertettük a tanulókkal a HTML dokumentum struktúrájával, felépítésével, a tag fogalmával, kezdődhet az érdemi munka, azaz a különböző tagok megismerése, használatuk elsajátítása.

A tagokat célszerű nyers dokumentumok formázásával tanítani. Olyan nyers szövegeket kell választani, melynél fokozatosan jönnek elő a formázáshoz szükséges formázóutasítások. Bevezető feladatnak, hogy a diákok is ráhangolódjanak a HTML dokumentum szerkesztésére, érdemes olyan szöveget választani, melynél a szövegtördelés mellett látványosabb szerkesztéseket is elvégezhetünk. Jól megválasztott szöveg esetén előjöhethet `<H1>...<H6>`, `<P>` tagok mellett a `<PRE>` és a `` tag is. Erre láthatunk példát az 1-es mellékletben, amely a C programozás alapjaival foglalkozik. A dokumentum bizonyos részeire használhatjuk a félkövér, `` és a dőlt, `<I>` formázóutasításokat.

Ennél a szövegnél kétféleképpen is meg lehet mutatni a tanulóknak, a programkód kialakítását – mivel két helyen is tartalmaz C programkódot –, melyre általános szabály, hogy a könnyebb olvashatóság érdekében azonos szélességű betűtípussal kell írni. Ezt megtehetjük egyrészt a `<PRE>` taggal, másrészt a `` tag `face` paraméterének segítségével.

A példánkban fontos kitérni a HTML nyelv speciális karaktereire. Feltehetjük a kérdést a tanulóknak, hogy vajon miért nem jelent meg a böngészőben a `<stdio.h>`? Tapasztalatom szerint a tanulók többsége érzi, hogy a kisebb, nagyobb jelekkel van valami baj. Miután sikerült megbeszélni a diákokkal a fennálló probléma okát, mutassuk meg nekik a megoldást, a HTML nyelv e speciális karakterekhez rendelt kódokat, az `<`-t és `>`-t. Előjáróban, nyugodtan beszélhetünk az ` `-ről is, mely később a táblázatoknál előkerül.

A szöveg színezésének köszönhetően tanulók láthatnak példát egymásba ágyazott tagokra, ezt kihangsúlyozandó említsük meg, hogy maga a legegyszerűbb HTML dokumentumnál, mely csak a struktúrát felépítő tagokból áll is ugyan erről az egymásba ágyazásról van szó. Ugyanis a tanulók sok esetben azokat nem tekintik tagoknak.

A tagok egymásba ágyazásánál a nyitó és záró tagok helyes sorrendjének elmagyarázását egyszerűen illusztrálhatjuk matematikai zárójelezés analógiájával, ahol a különböző zárójelek, különböző tagokat képviselnek:

```
<P><FONT color="#0000FF"> Ide kerül a <I>bekezdés</I> szövege.</FONT></P>
{ [ ( ) ] }
```

A szövegtördelés után, következhet egy picit nehezebben emészthető anyagrész, a táblázat készítés. Mielőtt elkezdenénk a diákoknak elmondani milyen tagokkal, és paraméterekkel lehet táblázatot létrehozni, rajzoljunk fel a táblára egy egyszerű táblázatot, és írjuk fel a kódját, hogy az alap tagok a diákok előtt legyenek, a paramétereket ráérünk azután elmondani, miután megértették a táblázat létrehozásának logikáját. Példának egy egyszerű 2x2-es táblázat megfelelő:

<pre><TABLE> <TR> <TD>1</TD> <TD>2</TD> </TR> <TR> <TD>3</TD> <TD>4</TD> </TR> </TABLE></pre>	<pre><TABLE> <TR> <TD> 1 </TD> <TD> 2 </TD> </TR> <TR> <TD> 3 </TD> <TD> 4 </TD> </TR> </TABLE></pre>
---	---

A megértést segíthetjük, ha magába a rajzba is beírjuk a táblázatot létrehozó tagokat, a megfelelő helyre, így jobban tudjuk szemléltetni az adott tagok funkcióját.

Miután leraktuk az alapokat, rögzíthetjük a tagok paramétereit megfelelő magyarázat mellett. A tagok működésének jobb megértése végett, mindenképpen térjünk ki a sor és cella formázóutasítások záró párjainak opcionálisára. Magyarázzuk el, hogy miért működik a kód akkor is, ha nem rakjuk ki azokat. Ettől függetlenül hangsúlyozzuk, hogy munkájuk áttekinthetősége érdekében rakják ki e tagok záró párját. Táblázatoknál másik fontos dolog,

amiről szót kell ejteni, hogy egy cella addig nem jelenik meg a böngészőben, amíg nem adunk neki valamilyen tartalmat. Utaljunk vissza az ` ` speciális karakterre, melynek segítségével üres cellát tudunk megjeleníteni.

A táblázat formázóutasításainál megtanulandó paraméterek közül, ami igazán magyarázatra szorul az a `colspan` és `rowspan`, ezért érdemes erre szintén mintát adnunk a tanulóknak.

```
<TABLE>
  <TR>
    <TD colspan="2">A</TD> <TD rowspan="2">B</TD>
  </TR>
  <TR>
    <TD>1</TD> <TD>2</TD>
  </TR>
</TABLE>
```

A		B
1	2	

A táblázat felrajzolása után, beszéljük meg a diákokkal, hogy hány sorról és hány oszlopról is van szó. Törekedjünk arra, hogy a HTML kódot a diákok adják meg, mi csak vezessük gondolatmenetüket, és adott esetben javítsuk ki őket.

Önálló feladatnak a diákok a 2. melléklet mintájához hasonlóan elkészíthetik saját órarendjüket. Ez azért jó, mert miközben dolgoznak, figyelhetjük munkájukat, ha valaki elakad munkájában, segíthetünk neki a többiek zavarása nélkül.

Azoknak a tanulóknak, aki fogékonyabbak az informatika iránt szorgalmi feladatnak feladhatunk egy viszonylag bonyolult táblázatot, amit otthon elkészíthetnek az órán tanultak alapján. Lássunk egy példát ilyen táblázatra:

A tananyagnak ez a része nem biztos, hogy a diákok kedvence. Ezért fontos, hogy az érdeklődésüket folyamatosan fenntartsuk, mindamelllett haladjunk is a tanulással. Következhet egy olyan szöveg, amelynek a formázását, a szöveg tördelését a diákok önállóan is elvégezhetik. Természetesen a szöveg tördelése ismétlés, és egyben gyakorlás számukra, szövegünkben törekedjünk arra, hogy valamilyen formában minden előkerüljön, amivel már

foglalkoztunk, de legyen benne újdonság is, amit a tanulókkal közösen nézzünk meg. Ilyen szövegre láthatunk példát a 3. mellékletben

A dokumentum tartalmaz szövegtördelést, előkerülnek a címsorok, a betűtípus, betű szín megváltoztatása és a táblázat is. Hasznos, ha megmutatjuk a tanulóknak, hogy táblázatot nem csak olyan esetben használhatunk, amikor ténylegesen oszlopokról, sorokról beszélünk, és adatainkat szeretnénk szegélyek közt megjeleníteni, hanem mint azt tettük a Word szövegszerkesztő esetében is, magának a szöveg formázása céljából is. Ilyenkor levesszük a táblázatról a szegélyt, és úgy alakítjuk ki a cellák elrendezését, ahogy a bennük lévő szöveget szeretnénk elrendezni a képernyőn. Ebben a példában a tartalomjegyzéket alakíthatjuk ki táblázat segítségével a <P> és
 tagok helyett.

Új anyagrészként a kép beszúrás és a hivatkozások kerülnek elő. Hivatkozások tanításánál ne csak a más, külső honlapra vagy dokumentumra való kapcsolatot mutassuk meg, hanem azt is hogyan lehet oldalon belül a könyvjelzők segítségével például tartalomjegyzéket létrehozni, amely oldalon belül megfelelő helyre ugrik a szövegben. Hivatkozások tanításánál nagyon fontos kitérnünk – még ha az éppen aktuális példában nem is szerepel – az abszolút és relatív hivatkozások közötti különbségre. Nem utolsó sorban pedig utaljunk vissza a HTML dokumentumok nevére, mely, mint azt már említettük nem szerencsés, ha szóközt vagy ékezetes karaktert tartalmaz.

Következő fontos anyagrész a listák – felsorolás és számozás - létrehozása. Ehhez érdemes olyan szöveget keresni, amelyben több helyen is felsorolást vagy számozást kell kialakítani az eddig tanult szövegtördelés mellett. Ilyen szövegre példa a 4. melléklet. Először írjuk fel a táblára a formázóutasításukat, majd mutassuk meg a szövegben a felsorolás kialakításának módját. Tapasztalataim szerint a tanulók ekkora már elegendő rutinnal rendelkeznek ahhoz, hogy az első példa után a szövegben található többi felsorolást önállóan is el tudják készíteni mindenféle nehézség nélkül. Egyvalami okozhat gondot néhány tanulónak, ez pedig az egymásba ágyazott listák, ezért ezt érdemes közösen elkészíteni a minta alapján.

Nagyon oda kell figyelni minden egyes tag tanításánál, hogy a tanulók tudjanak különbséget tenni tag, paraméter és paraméter érték között. Sajnos sok esetben keverik a fogalmakat. Talán mert nem tisztázódott bennük még az elején a különbség. Ezt mutatja az is, hogy arra a kérdésre, hogy "Melyik tag segítségével tudjuk a bekezdéseket igazítani?" még tudják a választ. Ellenben, ha úgy tesszük fel a kérdést, hogy "Melyik paraméter segítségével tudjuk a bekezdéseket igazítani?", akkor már vegyesen adnak <P> és align válaszokat.

Természetesen a mellékletben található szövegek az adott anyagrész bevezetéséhez, elsajátításához használhatók. Ahhoz viszont nem feltétlen elegendőek, hogy a tanulók rendszeren begyakorolják a HTML dokumentumok szerkesztését. Ehhez az szükséges, hogy több hasonló szövegen gyakorolják a tanultakat.

Formázandó szöveg választásánál, fontos szempont a szöveg tartalma. A tanárok ebben a kérdésben két táborra alkotnak. Vannak, akik nem informatikai tartalmú szöveg mellett érvelnek, míg mások – és magam is ide tartozom – pont amellet. Véleményem szerint az informatikának nagyon sok olyan területe van, ami nem fér bele az óra kereteibe, vagy azért mert nem érettségi követelmény, vagy pedig nem fér bele az időbe. Fontosnak tartom, hogy ettől függetlenül valamilyen formában mégis megmutassuk ezeket a területeket a diákoknak. Ezzel az olyan tanulók látókörét tudjuk szélesíteni, valamint érdeklődését növelni, akik fogékonyabbak az informatika iránt.

Miután átvettük az alap formázóutasítása, mely ténylegesen tananyag, rátérhetünk pár olyan tagra, amely ugyan nem a tananyag része, de mégis érdemes megtanítani. Ilyen például a <FRAMESET> és <FRAME> tagok, amelyek segítségével keretekre oszthatunk egy HTML dokumentumot, hogy majd az egyes keretekben más-más tartalmat jelenítsünk meg. A legjobb, ha készítnünk egy oldalt, ami összefogja az eddigi órai munkánkat. Nézzünk erre egy vázlatot:

Ide kerül külön HTML dokumentummal elkészített cím.	
Ide elkészíthetünk egy dokumentumot, melyben egy menüt alakítunk ki az eddigi órai munkáinkról.	Itt jelennének meg az órán elkészített dokumentumok, melyeket a menüből érünk el.

Mivel itt magára a keretek kialakítására kell összpontosítani, minden mást a már tanult tagokkal kell megoldani. Például a menüt elkészíthetjük listába ágyazott hivatkozásokkal. Ezt követően szintén extra anyagrészként a már elkészített összefoglaló dokumentumunkban a menüt lecserélhetnénk egy általunk készített képre, és megmutathatjuk a kliens oldali térképek létrehozását, használatát.

Végül, miután mindezeket megtanítottuk, mutassuk meg a diákoknak, hogyan tudunk elkészített dokumentumot az interneten közzétenni. Mutassunk be egy ingyenes oldalt, a regisztrálás menetével együtt, valamint az adott oldal használatát, a fájljaink feltöltésének módját. Mivel kezdetektől ez volt a cél, így amolyan hosszabb távú házi feladatnak feladhatjuk, hogy mindenki készítse el saját honlapját, és rakja fel az internetre is.

Típushibák

Az egyik legelőször előforduló hiba több tanulónál, amit nagyon sokáig elkövetnek a HTML tagok lezárásának elmulasztása. Miközben közösen formázunk velük valamilyen nyers szöveget, mindig emeljük ki szóban, hogy amint kirakunk egy tagot, azonnal zárjuk is le. Például a dokumentum struktúrájának felépítésekor, amint kitesszük a <HTML> tagot, azt ne a <HEAD> kövesse, hanem azon nyomban zárjuk le a </HTML>-el.

Szintén a tagok záró párjánál előforduló hiba, hogy bár kiteszik azt, de rosszul. Talán ez az egyik legtöbbször előforduló hiba, és szerencsére legtöbb esetben eléggé látványos eredményt ad, így könnyen megtalálható.

```
<FONT color="#CCCCCC">szöveg<FONT>
```

```
<FONT color="#CCCCCC">szöveg<FONT
```

Később mikor egyre több paraméter jelenik meg sokan hajlamosak a paramétert hibásan megadni. Ennek a hibatípusnak három fajtáját követik el. Egyikben teljes mértékben rosszul adják meg, mint ahogy azt már az előzőekben tárgyaltam:

```
<BODY> - bgcolor="blue"
```

Ez a fajta rossz paraméter megadás a füzetbe való jegyzetelés módja miatt van, ezért is fontos a kezdetekben az általános tag formáját felírni a táblára és munkánk során folyamatosan utalni rá. Később, miután minimális rutint szereznek a tanulók, ezt a fajta hibát elhagyják.

Másik fajta hiba, hogy amikor több paramétert kell megadniuk, vannak, akik az egyes paraméterekhez külön-külön létrehozzák az adott tagot. A `<BODY bgcolor="#00CC00" text="#CCCCCC">` esetében az alábbi formát használják:

```
<BODY bgcolor="#00CC00">
```

```
<BODY text="#CCCCCC">
```

Végül pedig előfordul, hogy a paraméter értékének megadásánál elfelejtik kirakni valamelyik idézőjelet:

```
<P align="justify>
```

Az egyes típushibákat érdemes bemutatnunk a tanulóknak a kivetítőn. Mutassuk meg nekik, milyen jelek utalnak a hiba helyére, hogyan lehet azokat gyorsan megtalálni a dokumentumukban.

Projektor használata

A HTML nyelv tanításánál nagyon hasznos eszköz lehet a projektor, feltéve, ha megfelelően használjuk azt. Nagyon sokat segít a tanulóknak – főleg kezdeteknél –, ha maguk előtt látják a szerkesztés folyamatát. Mindamellet előbb utóbb a diákoknak önállóan is tudniuk kell dokumentumot szerkeszteni.

Meg kell találnunk a megfelelő egyensúlyt, hogy az órán mennyi magyarázatot fűzünk a tanultakhoz, milyen gyakran és formában mutatjuk be konkrét példán keresztül projektor segítségével az egyes tagok működését, és természetesen időt kell hagynunk magunknak arra, hogy a diákok munkáját személyesen is ellenőrizzük.

Nehézségekbe ütközhetünk, ha mi ülünk a kivetített gép mellett, és párhuzamosan oldjuk a feladatot a diákokkal. Ugyanis nehéz megtalálni a megfelelő tempót, hogy mindenki tudja követni a szerkesztés menetét. Főleg, ha közben egy-két diáknak problémája akad, melyet egyedül nem tud megoldani. Ekkor választhatunk, hogy vagy felfüggesztjük addig a munkát, vagy gyorsan befejezzük az elkezdett szerkesztést. Mindkét variációnál van, ami mellette szól, és van, ami ellene. Van azonban egy amolyan köztes megoldás. Szerkesztés alatt nyugodtan ültessünk egy tanulót a kivetítőhöz, ezáltal elég, ha olyankor, amikor mi is ott ülnénk egyszerűen instrukcióval látjuk el, így több időnk marad figyelni az egész csoportra. Valamint abban az esetben, ha megfelelő diákot választunk erre a feladatra, a szerkesztés tempója is megfelelő lesz. Attól nem kell félni, ha esetleg a tanuló hibázik, ellenkezőleg. Ha sok tanuló által elkövetett típushibáról van szó, azt ott helyben meg lehet beszélni a

tanulókkal mi is volt a hiba, s hogyan lehet könnyen megtalálni a kódban. Tagadhatatlan tény viszont, hogy a kivetítőnél dolgozó tanuló munkájára nagyobb figyelmet kell szentelnünk, nehogy kivetítve maradjon bármilyen hiba.

Később használhatjuk a projektort arra is, hogy egyszerűen kivetítjük a minta dokumentumot, amit a diákoknak a forrásanyagból kell létrehozni. Így önálló munka során folyamatosan tudják ellenőrizni saját munkájukat a kivetített, megformázott dokumentum alapján. Ilyenkor persze akadhatnak olyan szerkesztési lépések melyeknél nem elég a kivetítés, hanem egyéb magyarázatra szorulnak, ezeket nyugodtan írhatjuk a táblára. Ezt a módszert használhatjuk gép előtti számonkérésnél is. Természetesen csak abban az esetben, ha a terem kialakítása megfelelő, minden tanuló látja rendesen, nehézség, forgolódás, felállás nélkül a kivetített képet. Ellenkező esetben jobb, ha a mintát papíron kapják meg.

Véleményem szerint ezt az anyagrészt, a HTML alapjainak megtanítását, információ interneten való elhelyezését legjobb 10. osztályban tanítani. Ekkor már a diákok megfelelő előismeretekkel rendelkeznek a rutinos munkavégzéshez. Tisztában vannak az operációs rendszer használatával, az azon végzendő műveletekkel. Mivel a HTML szerkesztésnek megvan a maga logikus sorrendje, – mint ahogy az a szövegszerkesztésnek is megvan – így a diákok előzetes szövegszerkesztési ismeretei szintén megkönnyíthetik a tanulást.

Kommunikáció az interneten

Manapság az internetet nem pusztán információ keresésére és elhelyezésére használják az emberek. Sokkal inkább kommunikációs eszközzé válik. Segítségével állandóan elérhetőek lehetünk. Az interneten való kommunikációnak sok formája van, melyet nem egyszerűen használunk, hanem már-már elképzelhetetlen nélkülük hétköznapijaink. E kommunikációs formáknak saját szabályai vannak, ezért fontos ezeket a tanulókkal is megismertetni.

Az egyik legelterjedtebb kommunikációs forma az internetes levelezés. Valójában a kommunikációnak ezt a formáját mindenki használja, aki ismeri az internetet és használja azt. Mégis kevesen vannak tisztába azon szabályokkal, melyeket figyelembe kellene venni egy e-mail megírásakor. Ezért is kell tananyag szintjén foglalkoznunk ezzel a területtel. Nézzük mik azok a szabályok, amiket mindenképpen érdemes ismertetni a tanulókkal.

Az e-mail legvonzóbb tulajdonsága, hogy gyors. Ezért – ok nélkül – nem illik partnerünket napokra megvárakoztatni a válasszal. Egy gyors üzenetre arról, hogy miért nem válaszolunk azonnal és részletesen, mindig van idő. Ugyanakkor a levelet írótól elvárható, hogy minél

rövidebben és érthetőbben próbálja meg összefoglalni a közlendőjét, mert az embereknek általában csak néhány percük van napközben az elektronikus postájuk átnézésére, s az egy-két képernyőnél hosszabb leveleket olvasatlanul elmentik, majd megfélemlenek róluk.

Egyesek hajlamosak a leveleik végére illesztett *sig block* megtervezésekor összes titulusaikat és munkahelyüket felsorolni, vagy sormintákkal, ASCII rajzokkal, idézetekkel, esetleg teljes életfilozófiájukkal kiegészíteni az aláírásukat. Ez sokszor fölösleges bosszúságot okoz a címzettnek. Próbáljuk ezt a részt minél kisebb terjedelműre megkomponálni (max. 4 sor), vagy készítsünk különböző részletességű szignókat, különböző alkalmakra. A hagyományos leveleknél a címzésre, megszólításra és aláírásra használt túlzottan hivataloskodó megoldások az elektronikus levelekben többnyire egyébként is mellőzhetők, sőt kerülendők.

A legtöbb levelezőprogramnál van olyan funkció, hogy ha egy levélre reagálni akarunk, akkor azt beilleszthetjük a válaszukba, jelezve, hogy mire akarunk hivatkozni. Ezt a fajta idézetet hagyományosan az eredeti szöveg sorai elé írt > karakter jelzi. Nem illendő ilyenkor a teljes levél idézése és visszaküldése, különösen akkor, ha hosszabb üzenetről van szó és mi csak egy egyszavas megjegyzést akarunk hozzáfűzni. Vegyünk annyi fáradságot, hogy kitöröljük a válaszuk szempontjából érdektelen sorokat és csak a fontosabb mondatokat idézzük.

Az e-mail használatával kapcsolatban gyakori kifogás a „*junk-mail*” elszaporodása, más emberek postaládájának elárasztása számukra érdektelen vagy redundáns információkkal. Ennek elkerülése végett tartsuk be a következőket: A levél témájának rovatába értelmes és informatív szöveget írjunk, mert így a címzett a levél beolvasása nélkül is el tudja dönteni, hogy kíváncsi-e rá.

Az e-mail postafiókok tulajdonosai felelősek annak kezeléséért. Rendszeresen olvassuk ki a postaládánk tartalmát, a fontosabb leveleket saját gépünkre töltjük le, a törlésre jelölteket időnként fizikailag is töröljük a szolgáltató gépről. Ne engedjük, hogy a postafiókunk beteljen a sok olvasatlan levél miatt, mert a további levelek feladói ilyenkor egy bosszantó hibüzenetet kapnak.

Munkánk során, többször is adhatunk a tanulóknak otthon elkészítendő szorgalmi feladatot, melyet nekik e-mailben kell elküldeni. Ezáltal ellenőrizhetjük mennyire tartják be az ide vonatkozó szabályokat.

Másik elterjedt kommunikációs forma az úgynevezett internetes csevegés, vagy talk. Erre több lehetőség is van. Egyrészt vannak külön olyan oldalak melyek ezt a szolgáltatást nyújtják, valamint vannak külön erre a célra írt programok is, mint például a Windows Live Messenger, vagy a Skype. Teljesen mindegy, hogy hol, milyen eszköz használatával csevegünk másokkal, a szabályok, melyeket illik betartani ugyan azok. Annál is inkább mivel a levelezéssel ellentétben interaktív párbeszédet folytatunk. Nézzük meg tehát ezeket a szabályokat is.

Használjunk vegyesen kis- és nagybetűt, mintha egy levelet vagy egy e-mail-t írnánk. Nem szerencsés, ha hosszabb szöveget küldünk egyszerre. Azáltal, hogy mondanivalónkat mondatokra tagoljuk, és úgy küldjük el a másoknak, megelőzhetjük ezt a problémát.

A beszélgetés végén mindig írjunk egy elköszönést és várjuk meg a másik oldal köszönését is. Különösen fontos ez, ha egy távoli valakivel kommunikálunk. Ne felejtsük el, hogy a kommunikációnk sebessége egyrészt a rendelkezésre álló sávszélességtől és a hálózat késéseitől is függ.

Legyünk tisztában azzal, hogy a talk megzavarhatja a másik embert éppen aktuális tevékenységében. A mai programoknál beállíthatjuk állapotunknak, hogy mennyire vagyunk elfoglaltak. Tartsuk tiszteletben ezeket az állapotjelzéseket.

Legyünk nagyon óvatosak, ha egyszerre több emberrel beszélgetünk, ugyanis ilyenkor komoly esély van rá, hogy előbb-utóbb összekeverjük, hogy kivel és miről beszélgetünk, ezáltal kérdéseinket és válaszainkat rossz helyre küldve komikus és/vagy kínos helyzetet teremthetünk.

Végezetül azért szót kell ejteni egy elég nagy problémáról, a biztonságról. Mindegy, hogy a kommunikációnak melyik formáját használjuk, vagy hogy kivel folytatjuk, mindig ügyelni kell arra, hogy személyes adatokat semmilyen formában ne közöljünk, ugyanis nem lehetünk biztosak abban, ki olvassa el rajtunk és a másik felén kívül az üzeneteket.

Összefoglalás

Munkámban rámutattam a XX. század második felének legfontosabb jelenségére, amit "informatikai forradalom"-ként, információs társadalom kialakulásaként tartanak számon. Elemeztem e jelenség összetevőit, az oktatásra, és az informatika oktatására való hatását, változtatásra kényszerítő erejét. A tanítási - tanulási folyamatban a stabil ismeretek megszerzéséről a képességfejlesztésre tevődik a hangsúly. Új, az információ közléséhez és feldolgozásához kapcsolódó képességek kialakítására van szükség, ugyanakkor átalakul a három alapkészség: az írás, olvasás és a számolás is. Ezekhez az igényekhez igazodik a Nemzeti Alaptanterv is, a magyar közoktatás alapidokumentuma.

Dolgozatomban külön fejezetet szántam a tanulók motiválásának, melyben megpróbáltam a lehető legjobb meghatározni a sikeres motivációt. Mindezt magának a motivációnak a részletes megvizsgálásával tettem. Meghatároztam azon külső és belső motivációkat, melyek ismeretében közelebb kerülhetünk magához a sikeres motivációhoz. Ebben a fejezetben törekedtem általánosan írni a témáról, így az nem pusztán az informatika tantárgy keretein belül lehet alkalmazni.

Ezt követően szemügyre vettem azon eszközöket, melyek használata nélkülözhetetlen a hatékony informatikaoktatásban, és természetesen azon belül is a weboktatásban. Külön foglalkoztam egy ma még nem igazán elterjed eszközzel, az aktívtáblával, s kifejtettem milyen lehetőségek rejlenek ebben az új eszközben.

Dolgozatom következő részében részletesen foglalkoztam a weboktatás különböző területeivel. Először a weboktatás alapjait képező azon hálózati ismereteket határoztam meg, melyek fontosak ahhoz, hogy az egész tananyag kerek egész legyen, arról nem is beszélve, hogy ezek az ismeretek nagyon hasznosak is a mai világban. Ezt követően – mivel ma az egyik legnagyobb információforrás az internet, ezért – az interneten való információ keresés megtanításának fontosságát emeltem ki. Majd egy sokkal részletesebb, és nehezebb területtel, az információ internetre való elhelyezésének módjával foglalkoztam. Mely természetesen magába foglalja magának az információnak az elkészítését, majd közzétételét. Megvizsgáltam milyen problémák merülhetnek fel ennek az anyagrésznek az elsajátításában, és hogyan lehet azokat hatékonyan orvosolni, valamint azt is, hogyan lehet a rendelkezésünkre álló eszközöket, mint például a tábla, vagy a projektor minél hatékonyabban felhasználni a tanításhoz. Végezetül az internetes kommunikációval, annak szabályaival foglalkoztam,

melyeket nagyon fontos ma minden internetet használó embernek ismerni, így e szabályok megtanítását a tananyagba is bevenni.

Azért választottam ezt a témát szakdolgozatom alapjául, mert úgy gondolom, az informatikának ez a területe az egyik legfontosabb. Egyrészt azért, mert rohamosan fejlődik, így fontos lépést tartani az újdonságokkal, másrészt mivel ma már nem is igazán tudjuk nélkülözni munkánk során. Mindezek ellenére, úgy gondolom, sok középiskolában nem foglalkoznak eléggé részletesen ezzel a témával.

Ezért is szerettem volna olyan munkát készíteni, mely nélkülözi az egyes területek részletes taglalását, inkább azoknak szól, akik az adott tananyagrészt, tanítani szeretnék, hogy tapasztalataimmal iránymutatást adjak nekik a tanításhoz. Elképzelhetőnek tartom, hogy ezt a célt nem sikerült olyan mértékben megvalósítanom, amennyire ténylegesen szerettem volna, ám úgy gondolom ez leginkább a tanítási rutin hiányának köszönhető. Mindemellett mégis úgy gondolom, hogy lehetőségek mellett, mégis egy a tanítás során használható, abban segítő munkát sikerült elkészítenem.

Köszönetnyilvánítás

Köszönetem szeretném nyilvánítani Dr. Juhász István témavezetőnek, aki munkámat koordinálta és tanácsaival, ötleteivel mindig, készséggel segített, valamint Izsépi Béla kollégámnak, aki szintén tanácsokkal és ötletekkel látott el.

Mellékletek

1. melléklet

A C programozás alapjai

A nyelv fejlesztése 1974-ben kezdődött el, **Dennis Ritchie** vezetésével. A B nyelv kibővítésére, hatékonyabbá tételére, felváltására készítették, ugyanis a B már nem volt elég hatékony az akkoriban megjelenő PDP-11 számítógép új UNIX-szának programozásához. Az amerikai szabványügyi hivatal 1983-ban hozta létre az ANSI C szabványosított, végleges változatát. Ezt a cikket azoknak ajánlom, akik már ismernek egy-két programozási nyelvet, de eddig még nehéznek tartották a C-t, most azonban szeretnének megismerkedni vele.

A C nem típusos nyelv, mert ugyan vannak adattípusai, az egyes adattípusok között automatikus adatkonverziók definiáltak. A legtöbb nyelvtől ezek csak abban különböznek, hogy a C-nek nincs előre definiált szöveg (string) adattípusa, karakterláncok tárolására karaktertömböket használ. A tömbökkel a későbbiekben részletesen foglalkozunk.

Annak érdekében, hogy a hagyományokat betartsuk, nézzünk egy "Hello Világ!" programot. A megjegyzéseket `/*` és `*/` fogja közre, illetve újabban már használható a `//`, viszont akkor ettől a két karaktertől az egész sor megjegyzésnek fog számítani.

```
#include <stdio.h>

//ez megjegyzés

/*de ez is*/

int main()
{
    printf("Hello Világ!");

    return 0;
}
```

Ez volna az első C programunk! A fordításhoz a *Dev-C++* valamelyik verzióját, vagy a *Turbo C 2.0*-t javaslom, ezek ingyenesek és letölthetők az Internetről. Nézzük hát végig mit tartalmaz. Az elején található `#include <stdio.h>` sor az előfordító számára közli, hogy mi a programunkban az `stdio.h` fájlban található alprogramokat kívánjuk használni. Az előfordító-utasításokra még részletesebben kitérünk. A következő sorban található a `main` függvény, ennek minden programban szerepelnie kell, ebben a függvényben kezdődik meg programunk futása. Ennek típusa többnyire `int`, de gyakran használják a `void`-ot is. Egy függvény típusa azt adja meg, hogy a visszatérési érték milyen típusú lesz.

A C-ben az alprogramokat mindig függvénynek nevezzük, ennek törzsét a { és } jelek fogják közre. A C blokkstruktúrált nyelv, és az egyes programblokkokat ezek a jelek fogják közre. A programban bárhova lehet blokkot elhelyezni oda, ahol egy utasítás szerepelhet. A következő sorban láthatjuk a printf függvényt, amely formázott adatkivitelét valósítja meg a képernyőn. Itt nem olyan formázásra kell gondolni, mint amit a szövegszerkesztőben használunk például, hanem arra, hogy a kivitelen nem csak szöveget, hanem különböző típusú adatokat is megjeleníthetünk. A függvény utolsó sorában található a return utasítás után található változó értékével tér vissza a függvény. Nem kötelező megadni, ha nem adjuk meg a visszatérési érték 0.

Hogy tudunk változókat deklarálni? Először megadjuk a típust, majd a változó nevét. Ha több azonos típusú változót kívánunk megadni, vesszővel választhatjuk el őket. Mint már említettem, a C nem használ string típust, ehelyett karaktertömböt (vektort) használ, aminek indexe megadja a karakterek számát. Figyelem! A C-ben a karakterláncok (sztringliterálok) NULL karakter végűek, azaz a \0 zárja le őket, tehát egy karaktert ez lefoglal. Számok tárolására legtöbbször az int (egész), double (lebegőpontos), real (valós) típusokat használjuk. Fontos még megjegyezni, hogy a legtöbb nyelvvél szemben a C-ben a többdimenziós tömbök megadásánál nem vesszővel választjuk el az indexeket, hanem külön szögletes zárójelben adjuk meg őket.

```
//A karaktertömbök:  
char szoveg[45];  
int a;  
int b;  
//Ez helyettesíthető így:  
int a,b;  
int a[45][23];
```

Ebben a részben még igen sokszor írtam, hogy később foglalkozunk vele. Ezt azért teszem így, mert az egyes részek az azokat megelőzőkre alapoznak. Tehát a viszontolvasásra, a következő számban.

Forrás:

```
<HTML>  
  <HEAD>  
    <TITLE>A C programozás alapjai</TITLE>  
  </HEAD>  
  <BODY>  
    <H1>A C programozás alapjai</H1>  
    <P align="justify">A nyelv fejlesztése 1974-ben kezdődött el,  
<B>Dennis Rithcie</B> vezetésével. A B nyelv kibővítésére,  
hatékonyabbá tételére, felváltására készítették, ugyanis a B már nem volt elég  
hatékony az  
akkoriban megjelenő PDP-11 számítógép új UNIX-szának programozásához. Az amerikai  
szabványügyi
```

hivatal 1983-ban hozta létre az ANSI C szabványosított, végleges változatát. Ezt a cikket azoknak ajánlom, akik már ismernek egy-két programozási nyelvet, de eddig még nehéznek tartották a C-t, most azonban szeretnének megismerkedni vele.</P>

<P align="justify">A C nem típusos nyelv, mert ugyan vannak adattípusai, az egyes adattípusok között automatikus adatkonverziók definiáltak. A legtöbb nyelvtől ezek csak abban különböznek, hogy a C-nek nincs előre definiált szöveg (string) adattípusa, karakterláncok tárolására karaktertömböket használ. A tömbökkel a későbbiekben részletesen foglalkozunk.</P>

<P align="justify">Annak érdekében, hogy a hagyományokat betartsuk, nézzünk egy "Hello Világ!" programot. A megjegyzéseket /* és */ fogja közre, illetve újabban már használható a //, viszont akkor ettől a két karaktertől az egész sor megjegyzésnek fog számítani.</P>

```
<PRE><FONT color="#0000FF">#include &lt;stdio.h&gt;</FONT>

<FONT color="#009900">//ez megjegyzés

/*de ez is*/</FONT>

<FONT color="#0000FF">int</FONT> main()

{

    printf(<FONT color="#0000FF">"Hello Világ!"</FONT>);

    <FONT color="#0000FF">return 0;</FONT>

}</PRE>
```

<P align="justify">Ez volna az első C programunk! A fordításhoz a <I>Dev-C++</I> valamelyik verzióját, vagy a <I>Turbo C 2.0</I>-t javaslom, ezek ingyenesek és letölthetők az Internetről. Nézzük hát végig mit tartalmaz. Az elején található #include <stdio.h> sor az előfordító számára közli, hogy mi a programunkban az stdio.h fájlban található alprogramokat kívánjuk használni. Az előfordító-utasításokra még részletesebben kitérünk. A következő sorban található a main függvény,

ennek minden programban szerepelnie kell, ebben a függvényben kezdődik meg programunk futása. Ennek típusa többnyire `int`, de gyakran használják a `void`-ot is. Egy függvény típusa azt adja meg, hogy a visszatérési érték milyen típusú lesz.</P>

`<P align="justify">`A C-ben az alprogramokat mindig függvénynek nevezzük, ennek törzsét a `{` és `}` jelek fogják közre. A C blokkstrukturált nyelv, és az egyes programblokkokat ezek a jelek fogják közre. A programban bárhova lehet blokkot elhelyezni oda, ahol egy utasítás szerepelhet. A következő sorban láthatjuk a `printf` függvényt, amely formázott adatkivitelét valósít meg a képernyőn. Itt nem olyan formázásra kell gondolni, mint amit a szövegszerkesztőben használunk például, hanem arra, hogy a kivitelen nem csak szöveget, hanem különböző típusú adatokat is megjeleníthetünk. A függvény utolsó sorában található a `return` utasítás után található változó értékével tér vissza a függvény. Nem kötelező megadni, ha nem adjuk meg a visszatérési érték 0.</P>

`<P align="justify">`Hogy tudunk változókat deklarálni? Először megadjuk a típust, majd a változó nevét. Ha több azonos típusú változót kívánunk megadni, vesszővel választhatjuk el őket. Mint már említettem, a C nem használ `string` típust, ehelyett karaktertömböt (vektort) használ, aminek indexe megadja a karakterek számát. Figyelem! A C-ben a karakterláncok (sztringliterálok) `NULL` karakter végűek, azaz a `\0` zárja le őket, tehát egy karaktert ez lefoglal. Számok tárolására legtöbbször az `int` (egész), `double` (lebegőpontos), `real` (valós) típusokat használjuk. Fontos még megjegyezni, hogy a legtöbb nyelvvél szemben a C-ben a többdimenziós tömbök megadásánál nem vesszővel választjuk el az indexeket, hanem külön szögletes zárójelben adjuk meg őket.</P>

```
<P><FONT face="Courier"><FONT color="#009900">//A
karaktertömbök:</FONT><BR>
<FONT color="#0000FF">char</FONT> szoveg[45];<BR>
<FONT color="#0000FF">int</FONT> a;<BR>
<FONT color="#0000FF">int</FONT> b;<BR>
<FONT color="#009900">//Ez helyettesíthető így:</FONT><BR>
<FONT color="#0000FF">int</FONT> a,b;<BR>
<FONT color="#0000FF">int</FONT> a[45][23];</FONT></P>
```

```
<P align="justify">Ebben a részben még igen sokszor írtam, hogy később  
foglalkozunk vele. Ezt azért teszem így, mert az egyes részek az  
azokat megelőzőkre alapoznak. Tehát a viszontolvasásra, a következő számban.<P>  
</BODY>  
</HTML>
```

2. melléklet

Órarend

	Hétfő	Kedd	Szerda	Csütörtök	Péntek
1	első óra	első óra	első óra	első óra	első óra
2	második óra	második óra	második óra	második óra	második óra
3	harmadik óra	harmadik óra	harmadik óra	harmadik óra	harmadik óra
4	negyedik óra	negyedik óra	negyedik óra	negyedik óra	negyedik óra
5	ötödik óra	ötödik óra	ötödik óra	ötödik óra	ötödik óra
6	hatodik óra	hatodik óra	hatodik óra	hatodik óra	hatodik óra
7	hetedik óra	hetedik óra	hetedik óra	hetedik óra	hetedik óra

Forrás:

```
<HTML>
  <HEAD>
    <TITLE> Órarend </TITLE>
  </HEAD>
  <BODY>
    <TABLE border="1" bgcolor="#00CC00">
      <CAPTION>Órarend</CAPTION>
      <TR bgcolor="#CCCCCC">
        <TH bgcolor="#00CC00">&nbsp;</TH> <TH>Hétfő</TH> <TH>Kedd</TH>
<TH>Szerda</TH> <TH>Csütörtök</TH> <TH>Péntek</TH>
      </TR>
      <TR>
        <TD>1</TD> <TD>első óra</TD> <TD>első óra</TD> <TD>első óra</TD>
<TD>első óra</TD> <TD>első óra</TD>
      </TR>
      <TR>
        <TD>2</TD> <TD>második óra</TD> <TD>második óra</TD> <TD>második
óra</TD> <TD>második óra</TD> <TD>második óra</TD>
      </TR>
      <TR>
        <TD>3</TD> <TD>harmadik óra</TD> <TD>harmadik óra</TD>
<TD>harmadik óra</TD> <TD>harmadik óra</TD> <TD>harmadik óra</TD>
      </TR>
      <TR>
        <TD>4</TD> <TD>negyedik óra</TD> <TD>negyedik óra</TD>
<TD>negyedik óra</TD> <TD>negyedik óra</TD> <TD>negyedik óra</TD>
```

```

        </TR>
        <TR>
            <TD>5</TD> <TD>ötödik óra</TD> <TD>ötödik óra</TD> <TD>ötödik
óra</TD> <TD>ötödik óra</TD> <TD>ötödik óra</TD>
        </TR>
        <TR>
            <TD>6</TD> <TD>hatodik óra</TD> <TD>hatodik óra</TD> <TD>hatodik
óra</TD> <TD>hatodik óra</TD> <TD>hatodik óra</TD>
        </TR>
        <TR>
            <TD>7</TD> <TD>hetedik óra</TD> <TD>hetedik óra</TD> <TD>hetedik
óra</TD> <TD>hetedik óra</TD> <TD>hetedik óra</TD>
        </TR>
    </TABLE>
</BODY>
</HTML>

```

3. melléklet

LINUX - UBUNTU

Tartalom

[A Linux születése](#)

[Ubuntu](#)

[Verziók](#)

A Linux születése

1991-ben Linus másodéves hallgatója volt a Helsinkii Egyetem számítástechnikai tudományok karának. Linus autodidakta hacker volt, saját operációs rendszert szeretett volna írni. Hogy miért? Mert a GNU/Hurd-re várhatóan néhány évet várni kellett volna. A 21 éves egyetemista a Intel 80386-os processzor védett módú (protected mode), feladat-váltó (task-switching) lehetőségeit szerette volna felfedezni. Ez körülbelül 1991 nyarának elején lehetett. A pontos dátumra maga Linus sem emlékszik. Egy biztos: egy e-mail tanúsága szerint 1991. július 3-án már a POSIX szabvány után érdeklődött az interneten, így ekkor már futhatott nála egy kezdetleges rendszer. A program fejlesztése a Tanenbaum-féle Minix alatt történt, eleinte Assembly nyelven.



1991. augusztus 25-én Linus egy történelmi levelet küldött a MINIX hírcsoportba (a levél magyar nyelvű fordítása):

"Üdv minden Minix-felhasználónak odakinn! Készíték egy (ingyenes) operációs rendszert (csak hobbi, nem lesz olyan nagy és profi, mint a GNU a 386- (486) AT-klónokhoz. Április óta kotyvasztom, és már kezd elkészülni. Szeretnék visszajelzéseket hallani, mit szerettek, illetve nem szerettek a Minixben, mivel az én operációs rendszerem némileg hasonlít rá (többek között azonos a fájlrendszer fizikai kiosztása gyakorlati okokból).

Mostanában ülttettem át a bash (1.08) és a gcc (1.40) programokat, és úgy tűnik, működnek a dolgok. Ez azt is jelenti, hogy pár hónapon belül valami használható fogok kapni, és kíváncsi lennék, milyen képességeket szeretnének az emberek. Minden javaslatot szívesen veszek, azt viszont nem ígérem, hogy meg is csinálom őket :-)

- Linus (torvalds@kruuna.helsinki.fi)

Ui.: Igen! Nincs benne Minix-kód és többszálú fs-sel (fájlrendszerrel) rendelkezik. Nem hordozható (portable)(a 386 feladatváltást használja stb.), és lehet, hogy soha nem is fog az AT-merevlemezeken kívül bármi mást támogatni, minthogy nekem csak ez van"

Valahogy így kezdődött a Linux története. Linus levele számos programozót megérintett. Andrew S. Tanenbaum - a MINIX operációs rendszer atyja - Hollandiában élő egyetemi tanár, operációsrendszer-kutató - egyik későbbi levelében azt írta, hogy a Linux elavult (Linux is obsolete), mert monolitikus, ezért a professzor nem jósolt neki nagy jövőt. A levélből óriási vita lett. Ennek ellenére a Linux az elmúlt 10 évben hihetetlen fejlődésen ment keresztül, és még mindig monolitikus.

Ubuntu

Debian "sid" alapú Linux disztribúció 6 hónapos kiadási ciklusokkal, a legfrissebb szoftverekkel, bugfixekkel és integrációs munkákkal, amelyek visszakerülnek a Debianba. Ez jellemzi az Ubuntu Linuxot.

Az Ubuntu szó afrikai eredetű, jelentése "*emberiesség másokkal szemben*". Az Ubuntu Linux ezt a filozófiát szeretné elterjeszteni a szoftver világban. Fejlesztői között több Debian és GNOME híresség is található. A disztribúció frissességét és az új dolgokra való reagálását jól jelzi, hogy ez volt az első olyan Linux disztró amely a GNOME 2.8-at szállította.

Az Ubuntu alapból GNOME-ot tartalmaz, de létezik egy KDE-s testvére a Kubuntu és egy XFce-re alapozott, a Xubuntu disztribúció. Mindhárom terjesztés azonos verzióit újratelepítés nélkül konvertálhatjuk át, vagy akár egyidőben mindhárom telepítve lehet. Ezt azért tehetjük meg, mert mindegyik disztribúció ugyanazon az archívumon alapul, csak a generált CD-k és a megjelenés (branding) különböznek.

Az Ubuntu aktuális, hosszú karbantartású stabil kiadása a "*Dapper Drake*" (Ubuntu 6.06 LTS), míg a legfrissebb stabil kiadása a "*Feisty Fawn*". A következő, 2007 októberében megjelenő kiadás kódneve "*Gutsy Gibbon*".

A projekt mögött anyagilag Mark Shuttleworth és cége, a Canonical Ltd. áll.

Verziók

[Aktuális stabil verzió](#)

[Aktuális LTS verzió](#)

Verzió	Kiadás időpontja	Kódnév
4.10	2004. október 20.	Warty Warthog
5.04	2005. április 8.	Hoary Hedgehog
5.10	2005. október 13.	Breezy Badger
6.06	2006. június 1.	Dapper Drake
6.10	2006. október 26.	Edgy Eft
7.04	2007. április 19.	Feisty Fawn
7.10	2007. október	Gutsy Gibbon

Forrás:

```
<HTML>
  <HEAD>
    <TITLE>LINUX - UBUNTU</TITLE>
  </HEAD>
  <BODY>
    <H1>LINUX - UBUNTU</H1>

    <H2>Tartalom</H2>
    <TABLE border="0">
      <TR>
        <TD><A href="#szuletetes">A Linux születése</A></TD>
      </TR>
      <TR>
        <TD><A href="#ubuntu">Ubuntu</A></TD>
      </TR>
      <TR>
        <TD><A href="#verziok">Verziók</A></TD>
      </TR>
    </TABLE>
    <H2><A name="szuletetes">A Linux születése</A></H2>
    <IMG src="Linus_Torvalds.jpeg" align="right" title="Linus Torvalds">
    <P align="justify">1991-ben Linus másodéves hallgatója volt a Helsinkii
Egyetem számítástechnikai
tudományok karának. Linus autodidakta hacker volt, saját operációs rendszert
szeretett
volna írni. Hogy miért? Mert a GNU/Hurd-re várhatóan néhány évet várni kellett
volna.
A 21 éves egyetemista a Intel 80386-os processzor védett módú (protected mode),
feladat-váltó (task-switching) lehetőségeit szerette volna felfedezni. Ez
körülbelül
1991 nyarának elején lehetett. A pontos dátumra maga Linus sem emlékszik. Egy
biztos:
egy e-mail tanúsága szerint 1991. július 3-án már a POSIX szabvány után érdeklődött
az
interneten, így ekkor már futhatott nála egy kezdetleges rendszer. A program
fejlesztése
a Tanenbaum-féle Minix alatt történt, eleinte Assembly nyelven.</P>

    <P align="justify">1991. augusztus 25-én Linus egy történelmi levelet
küldött a MINIX hírcsoportba (a levél
```

magyar nyelvű fordítása):</P>

<P align="justify"><I>"Üdv minden Minix-felhasználónak odakinn! Készíték egy (ingyenes) operációs rendszert (csak hobbi, nem lesz olyan nagy és profi, mint a GNU a 386- (486) AT-klónokhoz. Április óta kotyvasztom, és már kezd elkészülni. Szeretnék visszajelzéseket hallani, mit szerettek, illetve nem szerettek a Minixben, mivel az én operációs rendszerem némileg hasonlít rá (többek között azonos a fájlrendszer fizikai kiosztása gyakorlati okokból).

Mostanában ültetttem át a bash (1.08) és a gcc (1.40) programokat, és úgy tűnik, működnek a dolgok. Ez azt is jelenti, hogy pár hónapon belül valami használhatót fogok kapni, és kíváncsi lennék, milyen képességeket szeretnének az emberek. Minden javaslatot szívesen veszek, azt viszont nem ígérem, hogy meg is csinálom őket :-)

- Linus (torvalds@kruuna.helsinki.fi)

Ui.: Igen! Nincs benne Minix-kód és többszálú fs-sel (fájlrendszerrel) rendelkezik. Nem hordozható (portable) (a 386 feladatváltást használja stb.), és lehet, hogy soha nem is fog az AT-merevlemezeken kívül bármi mást támogatni, minthogy nekem csak ez van"</I></P>

<P align="justify">Valahogy így kezdődött a Linux története. Linus levele számos programozót megérintett. Andrew S. Tanenbaum - a MINIX operációs rendszer atyja - Hollandiában élő egyetemi tanár, operációsrendszer-kutató - egyik későbbi levelében azt írta, hogy a Linux elavult (Linux is obsolete), mert monolitikus, ezért a professzor nem jósolt neki nagy jövőt. A levélből óriási vita lett. Ennek ellenére a Linux az elmúlt 10 évben hihetetlen fejlődésen ment keresztül, és még mindig monolitikus.</P>

<H2>Ubuntu</H2>

<P align="justify">Debian <I>"sid"</I> alapú Linux disztribúció 6 hónapos kiadási ciklusokkal, a legfrissebb

szoftverekkel, bugfixekkel és integrációs munkákkal, amelyek visszakerülnek a Debianba.

Ez jellemzi az Ubuntu Linuxot.</P>

<P align="justify">Az Ubuntu szó afrikai eredetű, jelentése <I>"emberiesség másokkal szemben"</I>. Az Ubuntu Linux ezt a filozófiát szeretné elterjeszteni a szoftver világban. Fejlesztői között több Debian és GNOME híresség is található. A disztribúció frissességét és az új dolgokra való reagálását jól jelzi, hogy ez volt az első olyan Linux disztró amely a GNOME 2.8-at szállította.</P>

<P align="justify">Az Ubuntu alapból GNOME-ot tartalmaz, de létezik egy KDE-s testvére a Kubuntu és egy XFce-re alapozott, a Xubuntu disztribúció. Mindhárom terjesztés azonos verzióit újratelepítés nélkül konvertálhatjuk át, vagy akár egyidőben mindhárom telepítve lehet. Ezt azért tehetjük meg, mert mindegyik disztribúció ugyanazon az archívumon alapul, csak a generált CD-k és a megjelenés (branding) különböznek.</P>

<P align="justify">Az Ubuntu aktuális, hosszú karbantartású stabil kiadása a <I>"Dapper Drake"</I> (Ubuntu 6.06 LTS), míg a legfrissebb stabil kiadása a <I>"Feisty Fawn"</I>. A következő, 2007 októberében megjelenő kiadás kódneve <I>"Gutsy Gibbon"</I>.</P>

<P>A projekt mögött anyagilag Mark Shuttleworth és cége, a Canonical Ltd. áll.</P>

Verziók</H2>

<P>Aktuális stabil verzió

Aktuális LTS verzió</P>

<TABLE border="1">

<TH>Verzió</TH>	<TH>Kiadás	<TH>időpontja</TH>
<TH>Kódnév</TH></TR>		
<TD>4.10</TD>	<TD>2004. október 20.</TD>	<TD>Warty Warthog</TD></TR>
<TD>5.04</TD>	<TD>2005. április 8.</TD>	<TD>Hoary Hedgehog</TD></TR>
<TD>5.10</TD>	<TD>2005. október 13.</TD>	<TD>Breezy Badger</TD></TR>

```

        <TR><TD><FONT      color="#007700">6.06</FONT></TD>      <TD><FONT
color="#007700">2006.  június  1.</FONT></TD>  <TD><FONT  color="#007700">Dapper
Drake</TD></FONT></TR>
        <TR><TD>6.10</TD>      <TD>2006.  október  26.</TD>      <TD>Edgy
Eft</TD></TR>
        <TR><TD><FONT      color="#0000FF">7.04</FONT></TD>      <TD><FONT
color="#0000FF">2007.  április  19.</FONT></TD>  <TD><FONT  color="#0000FF">Feisty
Fawn</FONT></TD></TR>
        <TR><TD>7.10</TD>      <TD>2007.      október</TD>      <TD>Gutsy
Gibbon</TD></TR>
    </TABLE>
</BODY>
</HTML>

```

4. melléklet

Webcím

Tartalomjegyzék

- [URL](#)
- [Jellemzők](#)
- [Tartománynév](#)
 - [A tartománynevek felépítése](#)
 - [Legfelső szintű tartomány](#)
 - [Második szintű tartomány](#)

URL

A webcím, más néven URL (mely a Uniform Resource Locator [egységes erőforrás-azonosító] rövidítése), az Interneten megtalálható bizonyos erőforrások (például szövegek, képek) szabványosított címe. Először Tim Berners-Lee alkotta meg a World Wide Webben való használatra. A jelenleg használt formátumot részletesen leírja az IETF RFC 1738 szabványa.

A webcím az Internet történetének alapvető újítása. Egyetlen címben összefoglalja a dokumentum megtalálásához szükséges négy alapvető információt:

- a protokollt, amit a célgéppel való kommunikációhoz használunk;
- a szóban forgó gép vagy tartomány nevét;
- a hálózati port számát, amin az igényelt szolgáltatás elérhető a célgépen;
- a fájlhoz vezető elérési utat a célgépen belül.

Jellemzők

Egy tipikus, egyszerű webcím így néz ki:

`http://hu.wikipedia.org:80/wiki`

Ennek részei:

1. A `http` határozza meg a használandó protokollt. A protokoll neve után kettőspont (`:`) írandó.
2. A `hu.wikipedia.org` adja meg a célgép tartománynevét. Ez elé két perjel (`//`) írandó.
3. A `80` adja meg a célgép azon hálózati portszámát, amin kérésünket várja; ez elé kettőspont (`:`) írandó. Ezt a részt gyakran teljesen elhagyhatjuk, például esetünkben a `http` protokoll alapértelmezett portszáma a `80`.
4. A `/wiki` a kért elérési út a célgépen. Ez a rész mindig a perjellel (`/`) kezdődik.

A legtöbb böngésző nem is igényli, hogy a "`http://`" részt begépeljük egy weblap eléréséhez, hiszen az esetek döntő többségében úgylis ezt használjuk. Egyszerűen begépelhetjük a lap

címét, például: "hu.wikipedia.org/wiki/Bit". A főlap megtekintéséhez általában elég a tartomány nevét beírni, például "hu.wikipedia.org".

A webcímek egyéb részeket is tartalmazhatnak, http esetében például az elérési út után, egy kérdőjel (?) mögé helyezve keresési kérdés szerepelhet, ami egy get metódusú HTML űrlapból származik. Az elérési út után, attól egy kettős kereszttel (#) elválasztva szerepelhet a hiperszöveg egy részére hivatkozó azonosító. Ez az azonosító nem része a webcímnak, de gyakran szerepel vele kapcsolatban.

Tartománynév

Egy tartománynév (angolosan domainnév, illetve doménnév) egy számítógép egyedi neve az interneten, mely megkülönbözteti a többitől. A tartománynevek csupán címkek melyeket a nehezen megjegyezhető IP-címek helyett használhatunk.

A tartományneveket gyakran összekeverik a weblapokkal, webhelyekkel, a webcímekkel (URL) vagy az internetes szolgáltatásokkal, melyek az adott néven (pontosabban az adott névvel elérhető számítógépen) érhetőek el. A tartományneveknek nincs szoros köze a neveken elérhető szolgáltatásokhoz (mint ahogy a házsámoknak sincs például közük ahhoz, hogy a házban pékség vagy nyomda üzemel).

A tartománynevek felépítése

A tartománynevek rendszerének felépítése hierarchikus, vagyis a nevek részei közül egyesek alá vannak rendelve a név más részeinek. Hagyományosan a nevek a legalacsonyabb szintű elemmel kezdődnek, ami alá van rendelve az azt követő elemnek, ami az őt követőnek, és így tovább. Az elemeket ponttal választjuk el.

Példa

Jelen esetben legyen a példa a hu.wikipedia.org cím:

- Az org a legfelső szintű tartomány (TLD - top-level domain), mely alá általában szervezetek (angolul organization) regisztrálják neveiket.
- Ezalatt a wikipedia második szintű tartomány (SLD - second-level domain) helyezkedik el, mely általában a tartománynév tulajdonosára jellemző (cégnév, személynév, terméknév, védjegy stb.). Ez jelen esetben a Wikipédia.
- A hu az előbb említett wikipedia.org alá tartozó számítógépet jelöli, vagyis a "Wikipédia szervezet" egy "hu" nevű gépét. Ennek a résznek (tehát az SLD alá rendelt tagnak) a neve állomásnév (hostname, hostnév).

Mindezen tagok együtt (hu.wikipedia.org formában) kijelölnek egy számítógépet, melynek címe ebből a névből meghatározható, és így a gépen levő szolgáltatásokat igénybe lehet venni (például a rajta elérhető webszervert, ami a Wikipédia weblapok tartalmát szolgáltatja).

Legfelső szintű tartomány

A legfelső szintű tartomány (angolul top-level domain, röviden TLD) a tartománynevek (domainnevek, doménnevek) végződésére, az utolsó pont utáni részre utal. Például a wikipedia.org tartománynevben a legfelső szintű tartomány az org (vagy ORG, ugyanis a tartománynevek nem különböztetik meg a kis- és nagybetűket).

Kétféle legfelső szintű tartomány létezik. Az országkód szerinti legfelső szintű tartományt (ccTLD) országok vagy területek használják, és kétbetűsek (például Magyarországnak hu). Az általános legfelső szintű tartomány (gTLD) három- vagy több betűs és (legalábbis elviekben) bizonyos minősítésű szervezetek használják (például com az üzleti szervezetek esetében).

Második szintű tartomány

A második szintű tartomány (angolul second-level domain, röviden SLD) a tartományneveknek (domainnév, doménnév) a legfelső szintű tartományt megelőző része. (Köznapiban szóhasználatban ezt nevezik domain névnek.) Például a wikipedia.org tartományban a wikipedia a második szintű tartomány. A második szintű tartománynév mellett egy legfelső szintű használata kötelező.

A [Wikipédiából](#), a szabad lexikonból.

Forrás:

```
<HTML>
  <HEAD>
    <TITLE>Webcím</TITLE>
  </HEAD>
  <BODY>
    <H1>Webcím</H1>

    <H2>Tartalomjegyzék</H2>
    <UL>
      <LI><A href="#url" >URL</A></LI>
      <LI><A href="#jellemzok">Jellemzők</A></LI>
      <LI><A href="#tartomanynev">Tartománynév</A></LI>
    <UL>
      <LI><A href="#felepites">A tartománynevek
felépítése</A></LI>
      <LI><A href="#tld">Legfelső szintű tartomány</A></LI>
      <LI><A href="#sld">Második szintű tartomány</A></LI>
    </UL>
  </UL>
  <H2><A name="url">URL</A></H2>
  <P align="justify">A webcím, más néven URL (mely a Uniform Resource
Locator [egységes erőforrás-azonosító]
```

rövidítése), az Interneten megtalálható bizonyos erőforrások (például szövegek, képek) szabványosított címe. Először Tim Berners-Lee alkotta meg a World Wide Weben való használatra. A jelenleg használt formátumot részletesen leírja az IETF RFC 1738 szabványa.</P>

<P align="justify">A webcím az Internet történetének alapvető újítása. Egyetlen címben összefoglalja a dokumentum megtalálásához szükséges négy alapvető információt:</P>

-
 - a protokollt, amit a célgéppel való kommunikációhoz használunk;
 - a szóban forgó gép vagy tartomány nevét;
 - a hálózati port számát, amin az igényelt szolgáltatás elérhető a célgépen;
 - a fájlhoz vezető elérési utat a célgépen belül.

<H2>Jellemzők</H2>

<P>Egy tipikus, egyszerű webcím így néz ki:

http://hu.wikipedia.org:80/wiki

Ennek részei:</P>

-
 - A http határozza meg a használandó protokollt. A protokoll neve után kettőspont (:) írandó.
 - A hu.wikipedia.org adja meg a célgép tartománynevét. Ez elé két perjel (//) írandó.
 - A 80 adja meg a célgép azon hálózati portszámát, amin kérésünket várja; ez elé kettőspont (:) írandó. Ezt a részt gyakran teljesen elhagyhatjuk, például esetünkben a http protokoll alapértelmezett portszáma a 80.
 - A /wiki a kért elérési út a célgépen. Ez a rész mindig a perjellel (/) kezdődik.

<P align="justify">A legtöbb böngésző nem is igényli, hogy a "http://" részt begépeljük egy weblap eléréséhez, hiszen az esetek döntő többségében úgyszintén ezt használjuk. Egyszerűen begépelhetjük a lap címét, például: "hu.wikipedia.org/wiki/Bit". A főlap megtekintéséhez általában elég a

tartomány nevét beírni, például "hu.wikipedia.org".</P>

<P align="justify">A webcímek egyéb részeket is tartalmazhatnak, http esetében például az elérési út után, egy kérdőjel (?) mögé helyezve keresési kérdés szerepelhet, ami egy get metódusú HTML űrlapból származik. Az elérési út után, attól egy kettős kereszttel (#) elválasztva szerepelhet a hiperszöveg egy részére hivatkozó azonosító. Ez az azonosító nem része a webcímnak, de gyakran szerepel vele kapcsolatban.</P>

<H2>Tartománynév</H2>

<P align="justify">Egy tartománynév (angolosan domainnév, illetve doménnév) egy számítógép egyedi neve az interneten, mely megkülönbözteti a többtől. A tartománynevek csupán címkék melyeket a nehezen megjegyezhető IP-címek helyett használhatunk.</P>

<P align="justify">A tartományneveket gyakran összekeverik a weblapokkal, webhelyekkel, a webcímeikkel (URL) vagy az internetes szolgáltatásokkal, melyek az adott néven (pontosabban az adott névvel elérhető számítógépen) érhetőek el. A tartományneveknek nincs szoros köze a neveken elérhető szolgáltatásokhoz (mint ahogy a házzámoknak sincs például közük ahhoz, hogy a házban pékség vagy nyomda üzemel).</P>

<H3>A tartománynevek felépítése</H3>

<P align="justify">A tartománynevek rendszerének felépítése hierarchikus, vagyis a nevek részei közül egyesek alá vannak rendelve a név más részeinek. Hagyományosan a nevek a legalacsonyabb szintű elemmel kezdődnek, ami alá van rendelve az azt követő elemnek, ami az őt követőnek, és így tovább. Az elemeket ponttal választjuk el.</P>

<P>Példa

Jelen esetben legyen a példa a hu.wikipedia.org cím:</P>

Az org a legfelső szintű tartomány (TLD - top-level domain), mely alá általában szervezetek (angolul organization) regisztrálják neveiket.

Ezalatt a wikipedia második szintű tartomány (SLD - second-level domain) helyezkedik el, mely általában a tartománynév tulajdonosára jellemző (cégnév, személynév, terméknév, védjegy stb.). Ez jelen esetben a Wikipédia.

A hu az előbb említett wikipedia.org alá tartozó számítógépet jelöli, vagyis a "Wikipédia szervezet" egy "hu" nevű gépét. Ennek a résznek (tehát az SLD alá rendelt tagnak) a neve állomásnév (hostname, hostnév).

<P align="justify">Mindezen tagok együtt (hu.wikipedia.org formában) kijelölnek egy számítógépet, melynek címe ebből a névből meghatározható, és így a gépen levő szolgáltatásokat igénybe lehet venni (például a rajta elérhető webszerveret, ami a Wikipédia weblapok tartalmát szolgáltatja).</P>

<H3>Legfelső szintű tartomány</H3>

<P align="justify">A legfelső szintű tartomány (angolul top-level domain, röviden TLD) a tartománynevek (domainnevek, doménnevek) végződésére, az utolsó pont utáni részre utal. Például a wikipedia.org tartománynévben a legfelső szintű tartomány az org (vagy ORG, ugyanis a tartománynevek nem különböztetik meg a kis- és nagybetűket).</P>

<P align="justify">Kétféle legfelső szintű tartomány létezik. Az országkód szerinti legfelső szintű tartományt (ccTLD) országok vagy területek használják, és kétbetűsek (például Magyarországnak hu). Az általános legfelső szintű tartomány (gTLD) három- vagy több betűs és (legalábbis elviekben) bizonyos minősítésű szervezetek használják (például com az üzleti szervezetek esetében).</P>

<H3>Második szintű tartomány</H3>

<P align="justify">A második szintű tartomány (angolul second-level domain, röviden SLD) a tartományneveknek (domainnév, doménnév) a legfelső szintű tartományt megelőző része. (Köznapi szóhasználatban ezt nevezik domain névnek.) Például a wikipedia.org tartományban a wikipedia a második szintű tartomány. A második szintű tartománynév mellett egy legfelső szintű használata kötelező.</P>

```
<H5>A <A href="http://wikipedia.org">Wikipédiából</A>, a szabad  
lexikonból.</H5>  
</BODY>  
</HTML>
```

Irodalomjegyzék

Fogarasi Mihály – Pszichológiai ismeretek (főiskolai jegyzet, 4. fejezet)

Csapó Benő – Az iskolai tudás felszíni rétegei, Didaktika szöveggyűjtemény

Nemzeti Alaptanterv. - Bp.: Művelődési és Közoktatási Minisztérium (2007.)

Faragó Tamás – Multimédiás eszközök alkalmazása az informatika oktatásban (2007, szakdolgozat)

Brian W. Kernighan, Dennis M. Ritchie - A C programozási nyelv (1996.)

www.ph.hu/mf/23.07/21.html (2005.)

<http://kobakbt.hu/index.html> (2008.)

<http://www.webgobe.ro/ikomm/> (2008.)

<http://www.codexonline.hu/CodeX3/Alap/beginners/CBA.HTM> (2007.)

http://hu.wikipedia.org/wiki/Ubuntu_%28Linux-disztrib%C3%BAci%C3%B3%29 (2007.)

<http://hu.wikipedia.org/wiki/Url> (2007.)

http://hu.wikipedia.org/wiki/Bill_Gates (2007.)