

## Számítógépes problémamegoldás

*Digitális világunkban a tradicionális és a nem tradicionális számítógépes tevékenységek – programozás, valamint dokumentumkezelés és információszerzés – eltávolodni látszanak egymástól. Míg a tradicionális tevékenységek problémamegoldási módszerei inkább a mély, addig a nem tradicionális tevékenységek szinte kizárólagosan a kevésbé hatékony felületi metakognitív problémamegoldási megközelítések csoportjába tartoznak. Kutatások egyértelműen bizonyítják, hogy e kettősség és a felületi megközelítések széles körű elterjedése vezet a rendkívül nagyszámú hibás dokumentumokhoz, az ezek alapján született hibás következtetésekhez, a dokumentumok létrehozásához és feldolgozásához szükséges, indokolatlanul magas emberi és gépi erőforrásigényhez. Mindenképpen indokolt tehát a számítógépes tevékenységek, a számítógépes problémamegoldási módszerek rendszerbe foglalása. Létrehoztunk egy olyan egységes rendszert, amely magába foglalja a számítógépes problémamegoldáshoz alkalmazott valamennyi megközelítést, ugyanakkor hozzáilleszthető más tudományterületeken már elfogadott rendszerekhez. Felkutattuk az eddig csak izoláltan létező, a tradicionális számítógépes tevékenységeket leíró rendszereket, ezeket kiegészítettük a nem tradicionális számítógépes tevékenységekkel, végezetül e tevékenységeket elhelyeztük egy általánosan elfogadott problémamegoldási rendszerbe, mintegy kiegészítve és teljessé téve azt ezekkel az új tevékenységi formákkal.*

### Bevezetés

A számítógépek elterjedésével megváltozott a számítógépeket használók köre. Napjainkra nemcsak a múlt századra jellemzően, programozók használják a számítógépeket, hanem mindenki, nemre, korra, hozzáértésre tekintet nélkül. Folyamatosan piacra kerülnek újabb és újabb, kényelmesebbnél kényelmesebb eszközök – beleértve a hardver- és a szoftvereszközöket is –, melyek számítógépes gondolkodásra [44] gyakorolt hatását még nem látjuk tisztán. Mérések egyértelműen bizonyítják, hogy ezek az eszközök megváltoztatják gondolkodásunkat, [9], [24]. Kérdés azonban, hogy mennyire befolyásolható ez a folyamat, mennyire akarjuk ezt befolyásolni, mennyire vagyunk képesek, felkészültek erre.

A tanterveinkben megjelent a kulcskompetenciák között a „digitális kompetencia”, azt sugallva, hogy oktatási rendszerünk fel van készülve ennek fejlesztésére, tehát oktatási rendszerünk képes arra, hogy az informatika nyelvén értő tanulókat neveljen, [27], [28], [33], [34], [35]. Olyan tanulókat, akik ezzel a kompetenciával felvértezve képesek lesznek az eszközök hatékony használatára, képesek lesznek az eszközök hatásának befolyásolására. A legutóbbi felmérések eredményei azonban nem

ezt mutatják, [2], [13], [14], [15], [16], [37], [41], [42]. A megjelent tanulmányokból, amelyek a felmérések eredményeit közlik, illetve azokból, amelyek a számítógépekhez kapcsolódó új fogalmakat próbálják definiálni, az olvasható ki, hogy a digitális kompetencia fogalma még napjainkra sem egyértelmű, [17], [19], [31], [40], és ennek egyik következménye, hogy a digitális kompetencia egyenértékűvé vált az információszerzéssel és feldolgozással, elfeledkezve a tartalmak disszeminációjáról, és elfeledkezve a hatékony feldolgozáshoz szükséges számítógépes gondolkodásról.

### Helyzetkép

Az alacsony hatékonyságú számítógépes problémamegoldás világméretű probléma, [2], [13], [14], [15], [16], [37], [41], [42]. Az utóbbi közel másfél évtizedben több olyan mérési eredményt is publikáltak, amely azt fogalmazza meg, hogy a számítógépes ismeretszintek és a számítógépes problémamegoldási megközelítések újbóli definiálására, az eddigi megközelítések határozott és gyors átértékelésére van szükség, valamint ennek szellemében az oktatási rendszerünk átalakítására, [2], [3], [8], [13], [20], [21], [22], [23], [25].

Ha valaki azt hinné, hogy ezek csak más országok problémái, mivel mi már az 1995-ös NAT [33] megjelenése óta tanítunk számítástechnikát, informatikát, szemben sok-sok hozzánk képest elmaradott országgal, az nagyon téved. A magyar diákok számítógépes írástudásának, navigációs képességének tesztelése a PISA2009 Students On Line [36] felmérésben egyértelműen mutatja, hogy nagyon le vagyunk maradva. A tizenkilenc részt vevő ország közül alulról az ötödik helyen végeztünk. Még további pesszimizmusra ad okot azonban annak vizsgálata, hogy milyen kapcsolat van a tanulók PISA-eredménye és az iskolai számítógéphasználat között. Az átlagos 9 értékhez képest Magyarország -27-et ért el, úgy, hogy az utánunk következő legalacsonyabb érték -8, és a teljes tartomány a [-27; 42] intervallum. Határozottan nem dőlhetünk elégedetten hátra, várva a csodát.

### Számítógépes ismeretszintek

A 2013-ban megjelent IEEE&ACM jelentés [26] egyértelműen megfogalmazza a számítógépes tevékenységek egymásra épülő ismeretszintjeit:

- megértés (familiarity)
  - Ezen a szinten történik meg a koncepció felfedezése, annak megértése. A hangsúly a használat nélküli koncepció megértésén van. Arra a kérdésre ad választ, hogy „Mit tudsz erről a problémáról?”.
- használat (usage)
  - A második szinten a koncepció használata, alkalmazása történik konkrét esetekben. Magába foglalja az eszközök – például programok – koncepción alapuló helyes használatát, egy-egy speciális bizonyítási eljárás vagy elemzési mód használatát. Arra a kérdésre ad választ, hogy „Mit tudsz arról, hogy hogyan kell csinálni?”.
- megítélés (assessment)
  - Ezen a szinten a tanuló, a felhasználó képes a koncepció több szempontból történő átgondolására és/vagy mérlegelésére a probléma megoldásához. Az a szint már magasabb, mint a koncepció használata, mivel magába foglalja a megfelelő megközelítés kiválasztását a megértett lehetőségek közül. Arra a kérdésre ad választ, hogy „Miért csinálnád azt?”.

A számítógépes ismeretszintekhez [26] hasonlóan már 2006-ban megfogalmazásra kerültek a digitális írástudás szintjei [31]. A három fogalom leírását részletesen elemezve láthatjuk, hogy ezek nem mások, mint a számítógépes ismeretszintek egy

korábbi, kevésbé körülhatárolt megfogalmazásai. Ebben a megfogalmazásban is a megértés van a legalsó szinten, ezt követi a használat, majd harmadik szintre kerül az ismeretek integrálása, de a szerzők véleménye szerint az egyes szintek sorrendje nem követi a fontossági sorrendet, ami ellentmondáshoz vezet.

A digitális írástudás szintjei [31]:

- digitális kompetencia (digital competence)
  - A rendszer alapja a digitális kompetencia, amely tartalmazza a számítógépes ismereteket, megértést, attitűdöt és a készségeket. Feltételezi azonban, hogy a résztvevők magukra szedik (will draw upon) a digitális kompetenciát, el tudják dönteni, hogy milyen szintű digitális kompetenciára van szükségük az életvitelükhöz, és képesek a tudások folyamatos fejlesztésére, ha az szükséges (will draw upon digital competence as is appropriate to their life situation, and return to it as new challenges are presented by their life situation). A tanulmány szerzői azonban nem fogalmazzák meg egyértelműen, hogy mi szükséges a digitális kompetencia hatékony fejlesztéséhez, és ez a gyenge pontja.
- digitális használat (digital usages)
  - Második szintre kerül, de azt a szintet tekintve a szerzők a digitális írástudás központi és legfontosabb szintjének. A dokumentum megjelenése óta azonban egyértelművé vált, hogy ez a szint nem lehet a legfontosabb, talán ezzel magyarázható a 2013-as IEEE&ACM jelentés [26] létrejötte, fontossága.
- digitális transzformáció (digital transformation)
  - A digitális írástudás legmagasabb, harmadik szintje a digitális transzformáció, amelyet akkor érünk el, ha a digitális használat olyan szintre jut, amelyen már képesek vagyunk kreatív és innovatív számítógépes tevékenységek elvégzésére, és a használat során eljutunk a tevékenységek egy magasabb szintjére. Fontosnak tartjuk megjegyezni, hogy ez a szint azonban csak akkor érhető el, ha az első szintet valóban teljesítettük, mivel önmagában a második szint nem elegendő.

Talán a helyenként homályos megfogalmazásokkal magyarázható, hogy az egyébként értékes kutatás és eredményei nem váltak ismertté és elfogadottá. Hasonló ellentmondások fedezhetők fel a *Digital Literacy in Education* című kiadványban is [19].

Ezen ellentmondásos megfogalmazásokból is eredeztethető, hogy jelenlegi oktatási rendszerünk átugorja az első számítógépes ismeretszintet és a használatra helyezi a hangsúlyt. Ennek a szemléletmódnak az egyik következménye, hogy a végfelhasználók nem látják a problémát, a probléma megoldásához vezető algoritmust, csak kattintgatások nem tervezett sorozatával próbálnak eljutni egy eredményhez, amely nem feltétlenül az eredeti probléma megoldása. Tehát, nem tudják, hogy nem tudják. Tudatlanságuk megakadályozza őket abban, hogy tisztán lássák, hogy mit tudnak és mit nem. Ez a jól ismert Dunning&Kruger effektus [29]. Az első ismeretszint kihagyásának másik következménye, hogy a felhasználók így nem juthatnak el a harmadik szintre, ahol a többféle megközelítésből választhatnának, mérlegelhetnék azokat, megkeresve a leginkább optimális megoldásokat. Különösen veszélyes a felületi megközelítés egy olyan tudományterületen, amely gyorsan változik és folyamatosan jelennek meg az újabb eszközök és szolgáltatások, folyamatosan változnak a használati módok. Az ilyen tudományterületeken a jól meglapozott koncepció segítheti a felhasználókat a felületi változások feldolgozásában és az azokhoz történő gyors alkalmazkodásban.

A koncepció megértésének kihagyása vezethetett a rendkívül erős negatív korrelációhoz a magyarországi iskolai számítógép-ellátottság és a tanulói teljesítmények között a PISA2009 Students On Line mérésben [36].

## Célok

### ***Nem tradicionális számítógépes tevékenységek azonosítása***

Vizsgálataink azt mutatják, hogy a leginkább kidolgozott terület a számítógépes tevékenységek egy szeletét, a hagyományos értelemben vett programozásoktatást fedi csak le. Alig találni olyan kutatásokat, kutatási eredményeket, amelyek a nem programozók – végfelhasználók – számítógépes gondolkodásának fejlesztését, számítógépes problémamegoldási módszereit, megközelítéseit tanulmányozzák. Izoláltan felfedezhetők próbálkozások, de eddig még ezek rendszerezése nem történt meg.

### ***Számítógépes tevékenységek beillesztése a létező problémamegoldási rendszerekbe***

Kutatásaink során arra kerestük a választ, hogy melyek a lehetséges és az elterjedt számítógépes problémamegoldási megközelítések, melyek azok

a rendszerek, amelyek egyértelműen megfogalmazzák, hogy milyen problémamegoldási megközelítések szükségesek a hatékony számítógéphasználathoz, a számítógépes tevékenységek kivitelezéséhez.

### ***Számítógépes és nem számítógépes problémamegoldási megközelítések egységesítése***

Tanulmányunkban a különböző problémamegoldási módszerek, megközelítések rendszerezését és ezek számítógépes környezethez adaptálását végezzük el. Célunk egy olyan rendszer felállítása, amely egyrészt magába foglalja a már hatékonynak bizonyult rendszereket, valamint a programozói és felhasználói – nem tradicionális programozói – tevékenységeket is.

### ***Számítógépes problémamegoldási megközelítések rendszerezése***

A matematikában elterjedt, elfogadott Pólya-féle [38] problémamegoldási módszerek felismerhetők a programozásmódszertani megközelítésekben. Annak ellenére, hogy a számítástechnika, az informatika a matematikából nőtt ki, 1992-ig nem történt meg a programozási módszerek rendszerbe foglalása sem, nem beszélve más számítógépes tevékenységekről. Ebben az évben jelent meg Shirley Booth PhD dolgozatának könyvváltozata [7], a göteborgi egyetem gondozásában, azonban eredményei izoláltak maradtak, nem váltak széles körben ismertté és elfogadottá. Megítélésünk szerint óriási veszteség érte a számítógépes társadalmat, hiszen Booth rendszere egyértelmű kategóriákat definiál, amelyek leírják a programozási tevékenységek lehetséges megközelítéseit. Mindez történt több mint 20 évvel ezelőtt. Az azóta eltelt időszakban ilyen vagy ehhez hasonló rendszerszemléletű tanulmány, amely a számítógépes problémamegoldási megközelítéseket rendszerezi nem jelent meg. Olyan tanulmányok fellelhetők, amelyek a programozást tanulók probléma-megoldási szintjeit képesek mérni, felállítják a programozásoktatás, -tanulás során elért, elérhető szinteket [4], [5], [6], [30], [39], [41], de továbbra sem született olyan rendszer, amely valamennyi számítógépes tevékenységet magába tudja foglalni.

### ***Pólya problémamegoldási megközelítése***

Pólya [38], az általa megfogalmazott problémamegoldási megközelítésben négy szigorúan egymást követő fázist különböztet meg:

- A probléma megértése, annak meglátása, hogy pontosan mit kell elérjünk, hova kell eljutnunk.
- A terv megépítése, annak meglátása, hogy a különböző források hogyan kapcsolódnak egymáshoz és a várható kimenethez.
- A terv megvalósítása.
- Visszatekintés, az eredmény értékelése, tesztelése.

Pólya „csak” azt írja le, hogy melyik a jó, a hatékony problémamegoldási megközelítés. Nem bizonyított, nem hozott ellenpéldákat, nem végzett statisztikai elemzéseket, „csak” látta az azóta bizonyítottan hatékony módszert. Napjaink kvantitatív méréseket követelő világában lehet, hogy tapasztalatai nem kerülhetnének publikálásra, nem jutnának túl saját egyetemének falain. Szerencsére, nem ez történt, tapasztalatait leírta, publikálta, majd a következő nemzedék bizonyította, felfedezései nem merültek feledésbe.

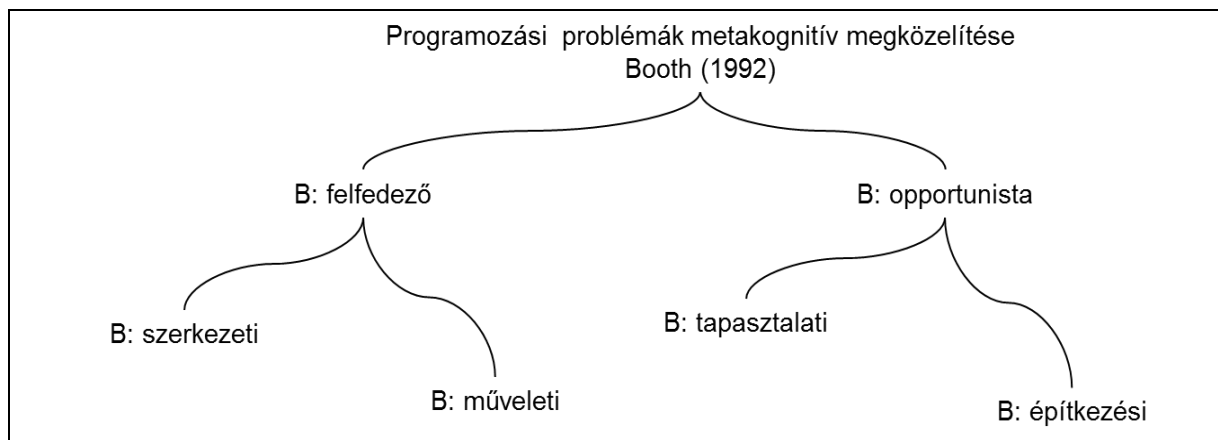
Mindenképpen érdemes megjegyezni, hogy a Pólya-féle problémamegoldási megközelítés lépései megfeleltethetők az IEEE&ACM jelentés [26] ismeretszintjeinek. Pólya első két szintje megfelel az ismeretszintek közül az elsőnek – koncepció megértése –, majd ezt követően egyértelmű a megfeleltetés.

### **A Booth-féle programozásorientált tevékenységek rendszere**

Booth 0 programozási problémák megközelítésére kidolgozott rendszere azonban, alig jutott túl a göteborgi egyetem falain, annak ellenére, hogy napjainkban is megállja a helyét és remekül beilleszthető a később publikált metakognitív problémamegoldási megközelítések rendszerébe.

Booth mérései alapján négy – egészen pontosan  $2 \times 2$  – kategóriát különböztet meg, amelyek képesek leírni a programozási problémák megközelítéséhez használt módszereket. A két nagyobb kategóriának az opportunista és felfedező (opportunistic & interpretative) megnevezést választotta, és ezeken belül megkülönböztet további két-két kategóriát:

- Az opportunista (opportunistic) kategóriába tartozó megközelítésekre jellemző a meglévő lehetőségek kihasználása, figyelmen kívül hagyva az eredeti problémát, a körülményeket, az elméleti hátteret, valamint a következményeket, eredményeket.
- A tapasztalati (expedient) alkategória a gyors és könnyű problémamegoldási megközelítéseket foglalja magába, amelyek eljuttatnak egy eredményhez, ami azonban nem feltétlenül az eredeti probléma megoldása. Az ilyen megközelítést alkalmazók automatikusan használják az eddig megtapasztalt módszereket.
- Az építkezési (constructual) alkategóriában a hangsúly a programozási nyelv eszközein van – a nyelven belüli szerkezetek, függvények, kulcsszavak –, amelyek szükségesek lehetnek a program megírásához.
- A felfedező (interpretative) kategóriába tartoznak azok a módszerek, amelyek a problémából, a bemeneti adatokból kiindulva próbálnak eljutni a feladat megoldásához.
- A műveleti (operational) alkategória esetén a fókuszban a probléma értelmezése és a program megírása, a probléma interpretálása a programozási keretek között. A problémát úgy tekinti, mint műveletek egy meghatározott sorrendjét, amelyeket a program végrehajt.



1. ábra A Booth-féle programozásorientált problémamegoldási megközelítések

- A szerkezeti (structural) megközelítésnél a probléma értelmezése és a program megírása, a probléma interpretálása a téma keretei között kerül a fókuszba. A probléma tartalmára fókuszál, és ezt alapul véve építi meg a programot. Ez a Pólya-féle klasszikus problémamegoldás.

### Case&Gunstone probléma megoldási megközelítései

Case&Gunstone [10], [11], [12] metakognitív problémamegközelítési rendszere a 2000-es évek elején látott napvilágot, felhasználva és rendszerezve a korábbi ilyen irányú tevékenységeket, elméleteket. Jelen szövegkörnyezetben fontosnak tartjuk hangsúlyozni, hogy ezek általános problémamegoldási megközelítések, és mint ilyenek figyelmen kívül hagyják vagy ismeretlen számukra Booth rendszere [7]. A megközelítéseket két nagy csoportba sorolják, hasonlóan Booth-hoz, azonban Marton&Saljo [32] elnevezéseit használják, mely szerint a különböző megközelítéseket „mélységük” alapján mély és felületi megközelítések csoportjába sorolhatjuk. Mindezek következtében Case&Gunstone egy mély és két felületi megközelítést különböztet meg (2. ábra). A mély megközelítést koncepcionális (concept-based) megközelítésnek nevezi. Ez a megközelítés nem más, mint a Pólya-féle problémamegoldási és a Booth-féle szerkezeti megközelítés. Kihagyja tehát a Booth-féle műveleti megközelítést. A felületi megközelítések csoportjának két kategóriája az algoritmikus és az információalapú megközelítések. Ez a két megközelítés megegyezik Booth programozási problémák megoldásához talált kategóriáival:

- mély megközelítés

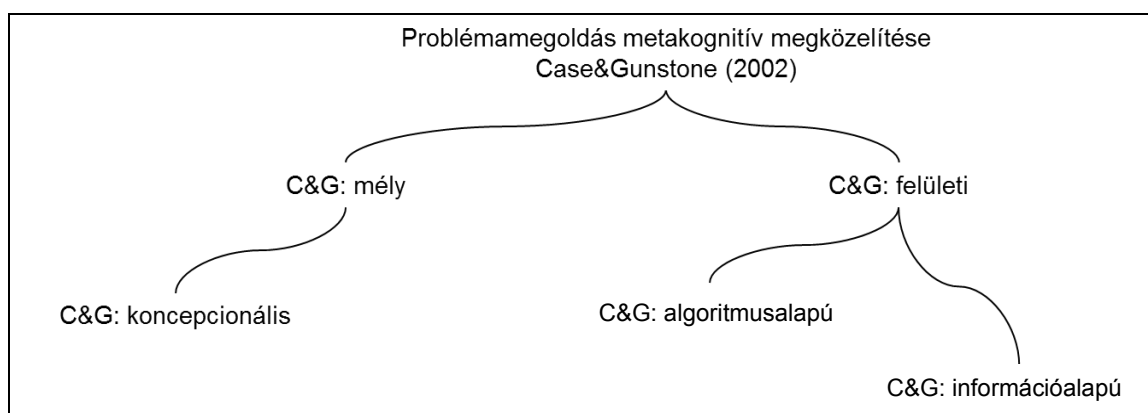
- Case&Gunstone (C&G): koncepcionális = Pólya-féle (P): problémamegoldás = Booth (B): szerkezeti
- felületi megközelítés
  - C&G: információ alapú = B: szerkezeti
  - C&G: algoritmus alapú = B: tapasztalati.

### Csernoch&Biró számítógépes problémamegoldási megközelítése

#### CAAD-alapú problémamegoldás

A számítógépes tevékenységek körének kiszélesedése azonban indokoltá tette Booth programozási problémák megoldásához használt rendszerének bővítését. Ebben a rendszerben a Booth által létrehozott műveleti kategóriát átneveztük úgy, hogy alkalmas legyen bármilyen számítógépes tevékenység algoritmikus megközelítésére. A Case&Gunstone rendszerében szereplő algoritmus alapú elnevezéstől, ami egy felületi megközelítés, azonban meg kell különböztessük ezt a mély megközelítést. Mindezen szempontokat figyelembe véve a kiterjesztett műveleti megközelítés a számítógépes algoritmus és tesztelés – Computer Algorithmic and Debugging (CAAD) – elnevezést kapta [18]. A CAAD-típusú megközelítés középpontjában az algoritmikus megközelítés áll: annak az algoritmusnak a megépítése, amely megadja, hogy a rendelkezésre álló inputból hogyan jutunk el a kívánt outputhoz. Az algoritmus megtervezését követi a kódolás, majd az eredmények diszkussziója, tesztelése, általánosítása [1].

A CAAD típusú megközelítések támogatják a számítógépes gondolkodás fejlesztését [44], valamint összhangban vannak a 2013-as IEEE&ACM [26] jelentésben rendszerezett számítógépes ismeret-szintekkel.



2. ábra A Pólya, Case&Gunstone-féle problémamegoldási megközelítés

TAEW-alapú problémamegoldás

A fentebb ismertetett rendszerek ugyanazon fogalmaknak más nevet választva, de hasonló szerkezetet hoztak létre. Booth műveleti megközelítése és Csernoch&Bíró által Case&Gunstone rendszerébe beillesztett CAAD-based megközelítés már alkalmas valamennyi mély megközelítésű számítógépes tevékenység leírására. Sem Booth, sem Case&Gunstone rendszerének kialakításakor azonban még nem léteztek azok a felhasználók és azok a felületek, amelyek elindítottak és teret engedtek egy újabb felületi megközelítés kialakításának. Azok a számítógépes problémamegoldási módszerek tartoznak ehhez a megközelítéshez, amelyek a grafikus felületek (GUI, Graphical User Interface) terv és cél nélküli használatát eredményezik. Az ilyen jellegű megközelítések leírására a próbálkozás és varázsló alapú – Trial-and-Error Wizard-based (TAEW) – elnevezést használjuk [18]. A megközelítés lényegét az alábbiakban foglaljuk össze.

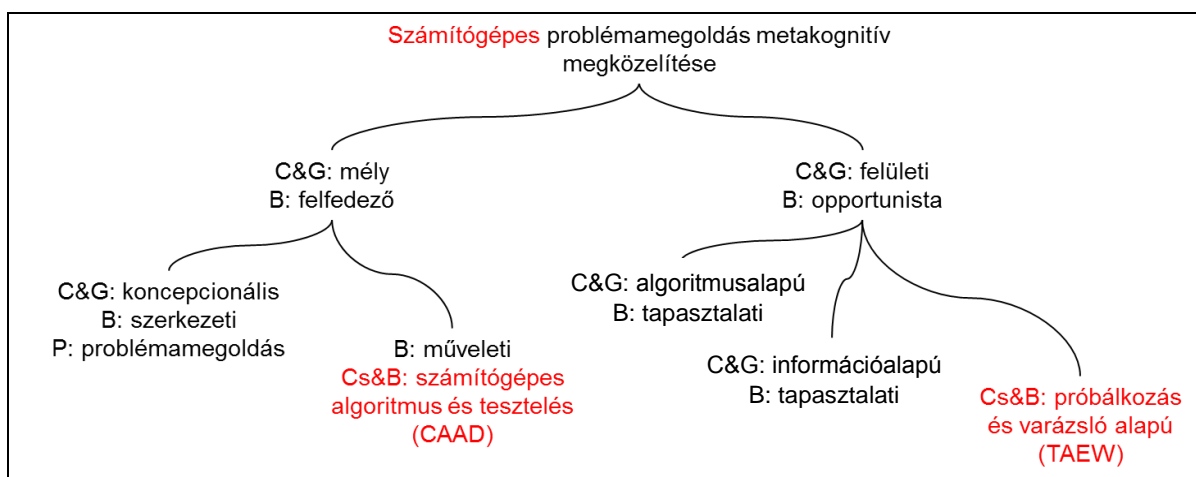
- A tevékenységsorozat végrehajtása tervezett lépéssorozat, algoritmus nélkül történik, amely folyamatban a felület használata a kódolás, gombokon és varázslókon keresztül. A szakirodalomban bricolage – barkácsolás – néven ismert ez a tevékenység. Fontos azonban megjegyezni, hogy mindenképpen indokolt volt a „bricolage” elnevezés pontosítása és helyettesítése egy egyértelmű kifejezéssel helyettesítése, mivel egyrészt jelentheti a „Mekk Elek”-féle barkácsolást is, valamint a konstruktivista szemlélet által támogatott önálló felfedezést is. A TAEW-alapú megközelítés a Mekk Elek-féle megközelítést takarja. Bármennyire is remekül

átadja a Mekk Elek-féle megközelítés a TAEW-típusú tevékenységek lényegét, nemzetközi viszonylatban azonban kicsi a valószínűsége, hogy Mekk Elek és tevékenysége ismert, így olyan megnevezésre volt szükségünk, amely bárki számára könnyedén azonosítható.

- A kapott kimenet nem feltétlenül a probléma megoldása.
- A felhasználó nem rendelkezik sem a képességgel, sem az igénnyel a kimenet helyességének ellenőrzésére.
- A felhasználó nincs tudatában annak, hogy az általa elvégzett tevékenység sorozat nem hatékony, óriási az emberi és gépi erőforrás, valamint idő igénye, ezentúl a kapott eredmény nem feltétlenül a feladat megoldása. Ebben a megközelítésben a hangsúly a felhasználó tudatlanságon van (Kruger & Dunning, 1999).

Összegzés

A különböző metakognitív problémamegoldási megközelítéseket összevetve azt tapasztaltuk, hogy több egymástól független rendszer is született, amelyek közel azonos eredményekre jutottak. Ezeket a megközelítéseket egységesítve és kiegészítve sikerült egy olyan komplex rendszert létrehozni, amely nemcsak a hagyományos tudományterületek problémamegoldási megközelítéseit képes lefedni, hanem a számítógépes tevékenységek, problémák megoldására használható megközelítéseket is. Mindezeket figyelembe véve, az előző rendszerek adaptálásával egy olyan bővebb rendszert hoztunk létre, amelyben meg tudunk különböztetni kettő mély és három felületi megközelítést.



3. ábra A számítógépes tevékenységek körének kiszélesedése indokoltá tette a korábbi problémamegoldási rendszerek egységesítését és bővítését

A mélyszerkezetű megközelítések lényege, hogy a probléma, a koncepció megértése az elsődleges tevékenység. A korábban született rendszerek összevetése során tapasztaltuk, hogy ezek a rendszerek izoláltan, de megfogalmazzák valamennyi mélyszerkezetű megközelítést. Hagyományos tudományterületeken és a tradicionális számítógépes problémamegoldásban – programozásban – ez már egy elfogadott megközelítés, míg a nem tradicionális számítógépes környezetekben sokkal inkább háttérbe szorul, és a felületi problémamegoldási megközelítések terjedtek el.

Nem tradicionális számítógépes tevékenységek közé soroljuk a végfelhasználók által végzett tevékenységeket, amelyek elsősorban dokumentumkezelést és információszerzést, feldolgozást jelentik. A felületi számítógépes megközelítések teljes leírásához azonban szükséges volt a már létező rendszerek bővítése. A bővítés szükségessége egyértelműen a grafikus felület megjelenésével és elterjedésével magyarázható. A végfelhasználók többsége a grafikus felület kezelését tekinti elsődleges tevékenységi formának, és ezek besorolhatóak a felületi megközelítés három alkategóriájának valamelyikébe, esetenként átfedések is lehetségesek. Koncepció hiányában ezek a tevékenységek cél nélküli kattintgatások sorozatában és az eredmények feltétel nélküli elfogadásában realizálódnak.

A felületi megközelítések következménye a hibás dokumentumok rendkívül magas száma a hibás dokumentumok alapján született hibás következtetések, ezek által okozott súlyos anyagi veszteségek, a dokumentumok előállításához szükséges rendkívül magas emberi és gépi erőforrás igény [37], [42]. Az informatikaoktatás céljainak az átalakítása mindenképpen szükséges és indokolt. Megérett az idő olyan módszerek fejlesztésére, adaptálására és bevezetésére, amelyek a mélyszerkezetű metakognitív problémamegoldási módszereket támogatják függetlenül a környezettől. Különösen fontos ennek a szemléletnek a követése a gyorsan változó tudományterületeken, így az informatikában. A koncepció megértése meg kell előzze a használatot, mivel a felületek folyamatos változása lehetetlenné teszi azok követését.

#### Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

A kutatást részben támogatta az OTKA K-105262 számú projekt.

#### Irodalom

- [1] BACKHOUSE, R. (2011) *Algorithmic Problem Solving*. Wiley. United Kingdom.
- [2] BEN-ARI, M. (1999) *Bricolage Forever!* PPIG 1999. 11th Annual Workshop. 5–7 January 1999. Computer-Based Learning Unit, University of Leeds, UK. Retrieved April 12, 2014 from <http://www.ppig.org/papers/11th-benari.pdf>.
- [3] BEN-ARI, M. & Yeshno, T. (2006) Conceptual models of software artifacts. *Interacting with Computers* 18, pp. 1336–1350.
- [4] BIGGS, J. (1979) Individual differences in study processes and the quality of learning outcomes, *Higher Education*, 8, pp. 381–394.
- [5] BIGGS, J. B. & Collis, K. E. (1982) *Evaluating the Quality of Learning: The SOLO Taxonomy*. New York: Academic Press.
- [6] BIRO, P. – CSERNOCH, M. (2013) Deep and surface structural metacognitive abilities of the first year students of Informatics. *Cognitive Infocommunications (CogInfoCom)*, 2013 IEEE 4th International Conference pp. 521–526, 2–5 Dec. 2013 doi: 10.1109/CogInfoCom.2013.6719303. <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6719303&isnumber=6719162>.
- [7] BOOTH, S. (1992) *Learning to Program. A Phenomenographic perspective*. Göteborg Studies in Educational Sciences 89. Acta Univesitatis Gothoburgensis.
- [8] BUDA, A. (2014) <http://www.slideshare.net/budaandras/buda-onk2014-albiz>. Letöltve: 2014. 11. 03.
- [9] CARR, N. (2011) *The Shallows: What the Internet is Doing to Our Brains*. W.W. Norton & Company, New York.
- [10] CASE, J. M. – GUNSTONE, R. F. (2002) Metacognitive development as a shift in approach to learning: an in-depth study. *Studies in Higher Education*, 2002, vol. 27(4), pp. 459–470.
- [11] CASE, J. M. (2000) *Students' perceptions of context, approaches to learning and metacognitive development in a second year chemical engineering course*, unpublished PhD, Monash University, Melbourne.
- [12] CASE, J. – Gunstone, R. – Lewis, A. (2001) *Students' metacognitive development in an innovative*

- second year chemical engineering course, Research in Science Education, 31, pp. 331–335.
- [13] CSERNOCH, M. (1997) Methodological Questions of Teaching Word Processing. 3rd International Conference on Applied Informatics: Eger-Noszvaj, Hungary, August 25–28, 1997. pp. 375–382.
- [14] CSERNOCH, M. (2009) Teaching word processing – the theory behind. Teaching Mathematics and Computer Science. 2009/1. pp. 119–137.
- [15] CSERNOCH, M. (2010) Teaching word processing – the practice. Teaching Mathematics and Computer Science. 8/2 (2010). pp. 247–262.
- [16] CSERNOCH, M. (2011) Clearing Up Misconceptions About Teaching Text Editing. Proceedings of ICERI2011 Conference, ICERI 2011, 4th International Conference of Education, Research and Innovation (2011. November 14–16., Madrid), ISBN 978-84-615-3324-4, Abstract CD ISBN 978-84-615-3323-7. # 1122. pp. 407–415.
- [17] CSERNOCH, M. – BIRÓ, P. (2014) Digital Competency and Digital Literacy is at Stake, ECER 2014 Conference, 1–5. September, 2014, Porto, Portugal.
- [18] CSERNOCH, M. – BIRÓ, P. (2014) Spreadsheet misconceptions, spreadsheet errors. Oktatáskutatók határon innen and túl. HERA Évkönyvek I., ed. Juhász Erika, Kozma Tamás, Publisher: Belvedere Meridionale, Szeged, 2014, pp. 370–395.
- [19] Digital Literacy In Education. (2011) UNESCO Institute for Information Technologies in Education. May 2011.  
<http://unesdoc.unesco.org/images/0021/002144/214485e.pdf>. Letöltve: 2014. 11. 03.
- [20] Erst denken, dann klicken I. Kein Kind ohne Digital Kompetenzen.  
[http://www.saferinternet.at/uploads/tx\\_simaterials/digitale-kompetenzen-handbuch-web-einzelseiten.pdf](http://www.saferinternet.at/uploads/tx_simaterials/digitale-kompetenzen-handbuch-web-einzelseiten.pdf). Letöltve: 2014. 11. 03.
- [21] Erst denken, dann klicken II. Konsumentenmildung in der Schule.  
[http://www.saferinternet.at/uploads/tx\\_simaterials/Erst\\_denken\\_dann\\_klicken\\_Konsumentenrechte\\_in\\_Internet.pdf](http://www.saferinternet.at/uploads/tx_simaterials/Erst_denken_dann_klicken_Konsumentenrechte_in_Internet.pdf). Letöltve: 2014. 11. 03.
- [22] GOVE, M. (2012) Michael Gove speech at the BETT Show 2012. Published 13 January 2012.  
<https://www.gov.uk/government/speeches/michael-gove-speech-at-the-bett-show-2012>. Letöltve: 2014. 11. 03.
- [23] GOVE, M. (2014) Michael Gove speaks about computing and education technology. Published 22 January 2014.  
<https://www.gov.uk/government/speeches/michael-gove-speaks-about-computing-and-education-technology>. Letöltve: 2014. 11. 03.
- [24] HAYLES, K. (2012) How we Think: Digital Media and Contemporary Technogenesis. The University of Chicago Press, Chicago, 2012.
- [25] ICT in schools survey – many children not getting what they need; teachers need more training and support (2013. április 19.) Az Európai Bizottsága sajtóközleménye, Brüsszel. URL:  
[http://europa.eu/rapid/press-release\\_IP-13-341\\_en.htm](http://europa.eu/rapid/press-release_IP-13-341_en.htm). Letöltve: 2014. 04. 12.
- [26] IEEE&ACM Report 2013. Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. December 20, 2013. The Joint Task Force on Computing Curricula Association for Computing Machinery (ACM) IEEE Computer Society.  
<http://www.acm.org/education/CS2013-final-report.pdf>. Letöltve: 2014. 11. 03.
- [27] Kerettanterv (2008) Magyar Közlöny, 20. sz. II. kötet, 2008. február 8.
- [28] Kerettanterv (2013) Letöltés:  
<http://kerettanterv.ofi.hu/index.html>. Letöltve: 2014. 11. 03.
- [29] KRUGER, J. – DUNNING, D. (1999) Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments. Journal of Personality and Social Psychology 77 (6): 1121–34.
- [30] LISTER, R. – SIMON, B. – THOMPSON, E. – WHALLEY, J. L., – PRASAD, C. (2006) Not seeing the forest for the trees: novice programmers and the SOLO taxonomy". = Proceedings of the 11th annual SIGCSE conference on Innovation and technology in computer science education, New York, NY, USA, 2006, pp. 118–122.
- [31] MARTIN, A. (2006) A European Framework for Digital Literacy. Digital kompetanse, 2-2006, pp. 151–161.
- [32] MARTON, F. – SALJO, R. (1984) Approaches to learning. = Marton, F., Hounsell, D.J. and Entwistle, N. J. (eds.), The Experience of Learning. Edinburgh: Scottish Academic Press, pp. 36–55.
- [33] NAT (1995) Nemzeti Alaptanterv 2007. Korona Kiadó 1995, Budapest.



- [34] NAT (2007)  
[http://janus.ttk.pte.hu/tamop/tananyagok/curriculum/a\\_nemzeti\\_alaptanterv\\_2007es\\_vltozat.html](http://janus.ttk.pte.hu/tamop/tananyagok/curriculum/a_nemzeti_alaptanterv_2007es_vltozat.html). Letöltve: 2014. 11. 03.
- [35] NAT (2012)  
[http://dokumentumtar.ofi.hu/index\\_NAT\\_informatika.html](http://dokumentumtar.ofi.hu/index_NAT_informatika.html). Letöltve: 2014. 11. 03.
- [36] OECD (2011) PISA 2009 Results: Students on Line: Digital Technologies and Performance (Volume VI).  
<http://browse.oecdbookshop.org/oecd/pdfs/free/9811031e.pdf>. Letöltve: 2014. 11. 03.
- [37] PANKO, R. R. (2008) What We Know About Spreadsheet Errors. Journal of End User Computing's. Special issue on Scaling Up End User Development. (10)2, pp. 15–21.
- [38] POLYA, G. (1954) How To Solve It. A New Aspect of Mathematical Method. Second edition (1957) Princeton University Press, Princeton, New Jersey. Magyarul: A gondolkodás iskolája, Gondolat Kiadó.
- [39] SHEARD, J. – CARBONE, A. – LISTER, R. – SIMON, B. – THOMPSON, E. – WHALLEY, J. L. (2008) Going SOLO to assess novice programmers, SIGCSE Bull., 2008, vol. 40 (3), pp. 209–213.
- [40] The Plurality of Literacy and its Implications for Policies. (2004) UNESCO Education Sector Position Paper. (p. 13),  
<http://unesdoc.unesco.org/images/0013/001362/136246e.pdf>. Letöltve: 2014. 11. 03.
- [41] TORT, F. – BLONDEL, F.M. – BRUILLARD, É. (2008) Spreadsheet Knowledge and Skills of French Secondary School Students. R. T. Mittermeir and M.M. Syslo (Eds.): ISSEP 2008, LNCS 5090, pp. 305–316, 2008. Springer-Verlag Berlin Heidelberg.
- [42] Van DEURSEN, A. – Van DIJK J. (2012) CTRL ALT DELETE. Lost productivity due to IT problems and inadequate computer skills in the workplace. Enschede: Universiteit Twente.  
[http://www.ecdl.org/media/ControlAltDelete\\_LostProductivityLackofICTSkills\\_UniversitofTwente1.pdf](http://www.ecdl.org/media/ControlAltDelete_LostProductivityLackofICTSkills_UniversitofTwente1.pdf), accessed 15-June-2014.
- [43] WHALLEY, J. L. – LISTER, R. – THOMPSON, E. – CLEAR, T. – ROBBINS, P. – KUMAR, P. K. A. – PRASAD, C. (2006) An Australasian study of reading and comprehension skills in novice programmers, using the bloom and SOLO taxonomies. = Proceedings of the 8th Australasian Conference on Computing Education - Volume 52, Darlinghurst, Australia, pp. 243–252.
- [44] WING, J. M. (2006) Computational Thinking. March 2006/Vol. 49, No. 3 Communications of the ACM.

Beérkezett: 2015. I. 7-én.

