

# ROBOTIKA A KÖZOKTATÁSBAN

## ROBOTICS IN PUBLIC EDUCATION

Kiss Róbert<sup>1</sup>

Kecskeméti Bányai Júlia Gimnázium, Kecskemét

---

### **Kulcsszavak:**

robotika  
algorithmikus gondolkodás  
innováció  
problémamegoldás  
közoktatás

### **Keywords:**

robotics  
algorithmic thinking  
innovation  
problem-solving  
education

### **Cikk történet:**

Beküldve: 2015. november 2.  
Átdolgozva: 2015. november 17.  
Elfogadva: 2015. november 30.

---

### **Összefoglaló**

*A robotika, a XXI. század vitathatatlanul egyik vezető kutatási területe. A középiskolai oktatásban is megvan a szerepe, hiszen a műszaki és informatikai kompetenciafejlesztés és problémamegoldás olyan eszköze lehet, amely megfelelő alapozást adhat a későbbi tudományos célok felé. Eszközként a MINDSTORMS robotok biztosítják az alapot.*

*Az elmúlt években egyre terjed Magyarországon is a közoktatásbeli robotika-oktatás. A nemzetközi gyakorlatban már több mint 10 éves hagyománya van a versenyeknek, amelyek 50-60 ország 20-30 ezer diákcsapatát mozgósítják. A nemzetközi versenykonceptiók sok esetben eltérnek a hagyományos magyarországi versenyektől, mind koncepciójukban, mind tartalmukban.*

*Ebben az írásban a középiskolai informatika oktatás megújítását szem előtt tartva a diákok algorithmikus gondolkodásának, problémamegoldási kompetenciáinak fejlesztésére mutatjuk be a robotika lehetőségeit. Mindezt összekapcsolva egy kontrollcsoportos vizsgálattal, amely alátámasztja az eszköz motivációnövelő hatását.*

### **Abstract**

*Robotics without doubt is one of the leading research fields of the 21st century. It can also play an important role in secondary education as a way of developing students' engineering, information technology and problem-solving competences, thus providing an adequate background for later ambitions in scientific fields. We rely on MINDSTORMS robots as appropriate supporting hardware.*

*Teaching robotics in IT lessons in public education has started to become popular in recent years. Concerning the international trends, competing in this field has had a tradition of more than ten years, involving 50-60 countries with approximately 20-30 thousand student teams. International tournaments often differ from the traditional contests in Hungary, both in concept and content. In this writing we demonstrate the potentials offered by robotics to develop students' algorithmic thinking and problem-solving abilities with a definite intention of reforming the teaching of information technology at secondary schools. The supplementary control-group trial is meant to prove how this method contributes to students' motivation in IT lessons.*

---

---

<sup>1</sup> e-mail cím: robekiss@gmail.com

## 1 Bevezető

A közeljövő technikai fejlődésének domináns irányai az informatika specializálódásán keresztül válnak valósággá. A XX. század science fiction regényeinek elképzelései közül egyre több inkább a tudomány, mintsem a fikció kategóriájába sorolható. A robotika az Isaac Asimov által megálmodott kitalációból hétköznapi gyakorlattá vált.

A műszaki tudományok oktatásának szükségessége nehezen vitatható, hiszen a háztartásokban jelenleg is számos olyan elektronikus eszköz működik, amely processzorvezérelt technológiára épül, a szórakoztató és kommunikációs elektronikától a fejlettebb háztartási gépekig. Ezeknek az eszközöknek az aránya egyre nő, és a jövő generációjának a gombok nyomogatásán túlmutató kompetenciákkal kell rendelkeznie, hogy ne váljon mindez a „varázslat” és „misztikum” eszközévé, hanem az egyszerű eszközhasználó is lássa és tudja a mögöttes algoritmusok emberi kreativitásban rejlő működését.

Az informatika oktatását olyan kettősség jellemzi, amely más tantárgyi struktúrákra nem jellemző. Egyrészt az elmélet, fogalmak, működési mechanizmusok, másrészt a gyakorlati használat kettőssége. A modern szoftverek fejlődésével egyszerű felhasználói szinten is lehetővé vált multimédiás anyagok szerkesztése és látványossá tétele. Az eszközök használhatóságának felhasználóbaráttá alakítása folytán egyre kevesebb iskolában megszerezhető tudásra van szükség a működtetéshez. A diákok a tantárgyak közül az informatikai eszközök használatához szükséges kompetenciákat sajátítják el leginkább iskolán kívüli keretekben. Az internet korlátlan szabadsága olyan információs körülményeket hozott létre kiegészítve a legegyszerűbb felhasználó számára is működtethető struktúrával, hogy iskolai oktatási rendszer nélkül is boldogulhat egy átlag felhasználó.

A kisebbség számára, akik az informatikával hivatásszerűen fognak dolgozni, ennél sokkal több kell. Mindennek az alapját nem csak technikai szinten, hanem még inkább a motiváció terén az általános- illetve középiskolában lehet lerakni.

A közoktatás informatika tanítása a legtöbb oktatási koncepció szerint alapvetően a gyakorlati géphasználton keresztül, az alapvető felhasználói szoftverek kezelésének tanításából, használatuk gyakorlásából áll. Háttérbe szorul az elmélet és az algoritmizálás, mint két régebbi terület. Nincs ezzel probléma, hiszen az átlagos felhasználó valóban leggyakrabban a szövegszerkesztés, valamint a hálózati kommunikáció (internet) lehetőségeivel találkozik és ezeket az eszközöket használja gyakorlati módon. Ugyanakkor az algoritmizálás (programozás) olyan kompetenciákat fejleszt, amely más tudományterületekre gyakorolt hatása nem elhanyagolható. Gondolok itt nemcsak a mindennapjainkat átható algoritmusokra, hanem a természettudományos gondolkodás absztrakt érvelésére és folyamatleírásaira. [2], [3]

Mindezen okokból az informatika oktatásának megújításához olyan eszközt kell keresni, amely használata egyszerűsége mellett magában hordozza a látványos és tanulói motivációt erősítő hatást.

## 2 Algoritmikus gondolkodás, algoritmizálás, programozás

### 2.1 Elméleti áttekintés

Az algoritmus olyan konkrétan meghatározott véges sok művelet sorrendje, amelyek végrehajtása elvezet a feladat (a probléma) megoldásához.

A problémamegoldás, amely során elemi részekre bontjuk, analizáljuk a feladatot, majd a megoldásnál a lényegi elemek kiemelésével keressük a megoldáshoz vezető utat, több különböző megközelítést is magában hordoz, amely elvezethet egy elfogadható eredményhez. Pólya György matematikai problémamegoldásra adott általános algoritmus is ilyen. Természetesen minél általánosabb a megfogalmazás, annál nehezebb az adott feladatra történő konkretizálás is.

A szakirodalom szerint az algoritmikus gondolkodásnak négy szintje különböztethető meg:

1. Már tanult eljárás emlékezetből történő előhívása, amely a deduktív gondolkodás alapja. Konkrét esetekben történő alkalmazása az általános algoritmus specifikálását igényli. Még nincs önálló algoritmus tervezés, csupán általános sémák konkretizálása.

2. Az előző szint bizonyos értelmű fordítottja. Az többszöri megfigyelésekből levont következtetések, általánosítások alapján az induktív gondolkodást használva, a lépéssorozat rögzítésére.
3. A tulajdonképpeni algoritmikus gondolkodás, amely során már tudatos törekvés zajlik a megfelelő algoritmus kiválasztására, megalkotására.
4. Kreativitás, amely során új algoritmus ötletek és lépéssorozatok készülnek.

A 1-4. szint esetén a különbség az algoritmus megalkotásához szükséges kognitív rendszer bonyolultságában rejlik.

Minden szinthez szükségesek azok az elemi lépések, amelyek egymásutánjára épülhet a megoldáshoz vezető út. Az elemi lépések szintaxisát és lehetőségeit az alkalmazás szempontjából releváns tudományterület specifikálja (sokszor több terület is közösen).

Az informatika oktatásának kezdeti szakaszában a programozás tisztán ilyen technikákat használt. Az absztraktsága miatt azonban kevés diák számára jelentett a sikeresen megírt algoritmus (program) katarzist. A felhasználói informatika térnyerésével ez a réteg tovább szűkült, hiszen sokkal egyszerűbb eszközökkel lehet látványosabb eredményt elérni. Nem jelent például motivációs kihívást a diákok számára a karakteres képernyőn megjelenő számsorban megszámlolni a páratlan elemeket, hiszen informatika órán alternatívaként a multimédiás elemek sokasága is megjelenik.

## 2.2 LOGO és LEGO

A hagyományos értelemben vett programozás, amely az algoritmusok tervezésének legabsztraktabb megközelítése, csak az emelt szintű érettségire felkészülés során jelenik meg. Mindezzel egyet lehet érteni, hiszen olyan speciális informatikai területről van szó, amely az átlag felhasználó számára fölöslegesnek tűnik a géphasználat során. A programozás által fejlesztett kompetenciák azonban nagyban hozzájárultak olyan gondolkodási módszerek, struktúrák kialakulásához, amelyek a műszaki, természettudományos és matematikai gondolkodás algoritmusait segítették. Az oktatási dokumentumokban továbbra is minden tudományterületen hangsúlyosan fejlesztendő kompetenciaként jelenik meg az algoritmikus gondolkodás.

Az általános iskolai informatikai gyakorlatban megjelenik az algoritmizálásnak egy olyan eszköze, amely a magas szintű programnyelvek sajátosságaival rendelkezik, és gyakorlati programozást tesz lehetővé. Ez a szoftver a LOGO, amelynek több szoftverfejlesztő által készített változata is forgalomban és használatban van. A tapasztalatok azt mutatják, hogy mindez 4-6. évfolyam tananyagában jelenik meg leginkább, ahol a szűkös órakeretek csak a vektorgrafikus ábrák rajzolását teszik lehetővé. A programmal egyszerűen és jól megtanítható az eljárások használata, akár egymásba ágyazva is. Nagyban segít a matematika szögekkel, arányokkal történő számításaiban, illetve a térbeli orientációt fejleszti. A programozási nyelvek legfontosabb sajátosságai közül az iterációk szerepe emelkedik ki. A változók használata és a feltételekhez kötött utasítás végrehajtás ugyan lehetőségként szerepel a programban, de tapasztalatok szerint nem jelenik meg a tanított anyagban. A probléma abban van, hogy a 10-12 éves korosztály számára a komplexebb alkalmazások elkészítése idegen a megszokott informatikai környezettől. Karakterkódolt utasítások egymásutánja messze van a megszokott „kattintok az egérrel” technikától. Az algoritmusok eredménye egy képernyőképet eredményez, ami ugyan lehet szép, de nem hordoz kellő motivációt, vagy gyorsan csökken a motiváltság. A program sajátosságai révén nagyon elkülönül a későbbiekben használt felhasználói szoftverektől, így 7-8. évfolyamra gyakorlatilag nyoma sem marad a megtanultaknak.

A karakteralapú programozás egyik már említett, és talán legfontosabb gátja a billentyűzethasználat és a karakter alapú programnyelvek szorosan kötött nyelvi szintaxisa. Néhány éve figyelhető meg az a jelenség, hogy a szokásos hardver elméleti oktatás kezdetén a számítógép elsődleges beviteli perifériájaként az egeret és nem a billentyűzetet említik a diákok. A programnyelvek beszélt nyelvektől eltérő nagyon szorosan kötött nyelvtani szerkezete érzékeny a gépelési hibákra, így a karakteralapú programozás tesztelési fázisa nagyobb részt a gépelési hibák javításából áll, ami megnehezíti az algoritmusok tervezését és motivációromboló hatásán keresztül rövid időn belül negatív tanulási rutint eredményez.

Seymour Papert, a LOGO nyelv megalkotója olyan eszközt hozott létre, amely nagyszerűen alkalmas volt algoritmikus gondolkodás fejlesztésére, de az informatikai eszközök fejlődése továbblépésre kényszerít. [5]

A kreativitás fejlesztésére vitathatatlanul egy olyan eszköz talán a legalkalmasabb, amely története során játékként indult, de az oktatás hamar felismerte a benne rejlő lehetőségeket. A LEGO a konstrukció ötletessége és a mögöttes mérnöki koncepció miatt továbbra is gyakorlatilag egyeduralgoló a területen. Részben Papert ötletei nyomán kiegészítve mindezt egy komoly informatikai háttérrel olyan eszköz jött létre, amely alkalmas arra, hogy a legmodernebb technológiát megjelenítve, az algoritmikus gondolkodás fejlesztéséhez olyan eszközt adjon, amely minden hibáját kiküszöböli a fentebb vázolt rendszereknek, ugyanakkor lehetővé teszi a korábban említett elvárásoknak megfelelést. A létrehozott eszköz egy programozható minikomputer, amely szenzorain keresztül érzékeli környezetét, megírt programja segítségével reagálhat rájuk, és a LEGO kreativitását használva csak az emberi fantázia szab határt a benne rejlő konstrukciós lehetőségeknek. Ez az eszköz a MINDSTORMS® névre keresztelt robot. (Jelenleg forgalmazott két változata: NXT és EV3.)

### 3 MINDSTORMS® robotok

Seymour Papert gondolata alapján is érdemes megvizsgálni a LEGO cég MINDSTORMS® konstrukcióját. „A játék irányítja-e a gyereket, avagy a gyerek a játékot?” Az eszköz egyértelműen csak a lehetőségeket biztosítja a kreatív fantázia számára. A robot csak arra képes, amit a felhasználó által megírt programja enged. Ez egy fontos gondolat és mindennek a tudatosítása a diákokban az eszközhasználat során érhető el.

A robot működését tekintve úgy képzelhető el, hogy készítünk egy konstrukciót, általában a célfeladatra, mindehhez LEGO építőelemeket használva. Beépítjük a konstrukcióba a robot agyát képező minikomputert, amelyhez szervo motorokat és a környezet fizikai, kémiai jellemzőit mérni tudó szenzorokat kapcsolhatunk. Egy számítógépen elkészítjük a robot programját, amely a szenzorokból érkező adatokat értelmezi és az utasításoknak megfelelően vezérli a motorokat, vagy a konstrukciót valamilyen válaszadásra kényszeríti. Az elkészült programot a robotra feltöltve, az már autonóm módon képes a működésre a programjának megfelelően. A működés tehát függ a megépített hardvertől és a megírt programtól. Mivel a robot viselkedését befolyásolja a környezet, ezért hibás működés esetén vagy a konstrukciót, vagy a programot kell átalakítani. Ez nem egyszerű probléma, hiszen a működést a környezet teljes hatásmechanizmusa befolyásolja, amelyből a szenzorok csak néhányat tudnak érzékelni. Például ugyan mérhető a szenzorra jutó fényintenzitás, de mivel ez nem homogén, ezért az árnyékok zavarhatnak, csakúgy, mint mozgás esetén a felület tapadási együtthatója, amely biztosan nem egyenletes. Hibás működésnél a döntés, hogy a programot vagy a hardvert változtassuk-e, már a problémamegoldás tárgykörébe esik. A hibakeresés a problémamegoldás lényeges összetevője. Nem véletlenül ez a PISA mérések problémamegoldás területének egyik hangsúlyos eleme. A robot nem várt viselkedését figyelve, annak azonosítása, hogy tíz esetből egy alkalommal produkált jelenséget mi okozza, nagyon nehéz. Ez a természettudományos kísérlet-megfigyelés-adatelemzés módszeréhez is közelebb visz.

Az eszközről további magyar nyelvű információk találhatóak a [4], [8], [9], [10], [11], [12] hivatkozásban megjelölt művekben.

### 4 Robotok használata az oktatásban – kontrollcsoportos kísérlet

A MINDSTORMS® robotok használatának oktatásbeli lehetőségeit érdemes megvizsgálni a gyakorlatban is. Bár a leírtak meggyőzőek lehetnek, de a tényleges alkalmazás világíthat rá a problémákra. Az alkalmazás a közoktatásban kétféle szervezésben képzelhető el. Az egyik a szakköri forma, amelynél csupán az érdeklődők vesznek részt a használatban és a tanulási folyamatban. A másik út a tanórai alkalmazás. Ebben az esetben minden diák részt vesz a folyamatban.

A 2009/2010-es tanévben egy kontrollcsoportos kísérletben vizsgáltuk, hogy az eszköz alkalmas-e az algoritmikus gondolkodás fejlesztésére és a tanulói motiváció megerősítésére.

A kísérlet időintervalluma a tanév II. féléve (február-május) volt, informatika tanóra (heti 1 óra) keretében. A kísérleti intervallum rövid, de a tantervi struktúra ezt tette lehetővé, valamint a szakköri formában már működő koncepció csupán megerősítést igényelt.

A kontroll csoport két 16 fős, 6. évfolyamos (12 éves) csoport, LOGO (Imagine) programozási nyelvet használt a kísérleti időszakban.

A kísérleti csoport két 16 fős, 6. évfolyamos csoport, LEGO Mindstorms® NXT robotokat használt, NXT-G ikon alapú keretprogrammal. Két tanulóként egy-egy eszköz használatát tudtuk biztosítani, így a csapatban dolgozás, a team munka is előtérbe került.

A kísérlet eredményességét saját fejlesztésű mérőeszközökkel vizsgáltuk.

A mérési koncepció szerint a kísérlet kezdetén, az algoritmikus gondolkodás fejlettségét mérő környezetfüggetlen, 45 perces papír-ceruza tesztet használtunk. A fejlesztési periódus végén, a bementi méréssel ekvivalens tesztet, valamint attitűdöket, motivációt vizsgáló kérdőívet írtak meg a kísérletben részt vevő tanulók.

Vizsgálati szempontokként a hozzáadott értéket mértük a programozói tudás alapjai és komplexitása terén, és az algoritmikus gondolkodás fejlettségére vonatkozóan, valamint az attitűdök és motiváció alakulását a kísérleti periódus során.

A papír-ceruza teszt összeállításának nehézsége abban rejlett, hogy a célnak megfelelően az algoritmikus gondolkodás fejlettségét, összetevőinek működését szeretnénk volna feltérképezni. A megvizsgált szakirodalom nem tartalmazott utalást olyan vizsgálatokra, mérésekre, amelyek kiindulási alapot jelenthettek volna.

A kísérleti és kontroll csoportban használt két programnyelvi környezet alapvetően eltér egymástól. Mindkettő az életkori motivációt kihasználva a látványelemekre épít, de míg a LOGO alapvetően karakter alapú (karakterek gépelésével lehet a forráskódot előállítani), addig az NXT-G egérhasználatra épül.

Más alapvető filozófiai különbség is van a két rendszer között. A robotprogramozás a szenzorhasználatból eredően, a környezet hatásaira történő reagálást helyezi előtérbe. Vagyis a programok nagymértékben épülnek a robot épített környezetben megjelenő hatásokra. Ezek sokszor csak a robot környezetbe helyezésével válnak ismertté, így a program felkészítése ezekre a hatásokra csak általános. A LOGO esetében a paramétereket a programozó adja meg az esetek többségében, és a környezeti hatás ezért kisebb és kevésbé realizisztikus.

Az LOGO Imagine programot a tanórán (a korosztály esetében) elsősorban különböző geometriai ábrák rajzolására használták a tanulók. Ezen ábrák szimmetriatulajdonságait kihasználva lehet a forráskódot megtervezni. Természetesen a fejlesztőkörnyezet alkalmas arra, hogy a magas szintű programnyelvek minden fontosabb eszközét használjuk, de ezek alkalmazása motivációs és időbeli gátakba ütközik.

Az elkészítendő tesztel szemben azt vártuk el, hogy egyik programnyelvet se favorizálja a másikkal szemben, így lehetőleg objektív képet adjon a mért területen.

### 3.1 A minta elemszámai

A kísérlet során három alkalommal került sor papír-ceruza mérésre, így a mintaelemszámok a tanulók hiányzásai miatt kis mértékben különböznek. Az adatok elemzésekor a három adathalmaz illesztésénél a legkisebb mintaelemszám az, amelynél az adatsor hiánytalan.

1. táblázat. A minta elemszámai a kísérletben részt vett csoportok szerinti bontásban (fő).

	Nyitó teszt	Záró teszt	Kérdőív
Kontroll csoport	33	33	30
Kísérleti csoport	26	32	32
Teljes létszám	59	65	62

A kísérlet tervezésénél arra törekedtünk, hogy a kísérleti és kontroll csoport létszáma körülbelül azonos legyen. A gimnáziumban működő két párhuzamos osztály profilja a tanult idegen nyelv vonatkozásában tér el (angol nyelv, illetve német nyelv). Mivel a robotprogramozásnál használt szoftver angol nyelvű, ezért mindkét osztályon belül fél-fél csoport vett részt a kísérleti és kontroll elrendezésben.

### 3.2 Nyitó mérés

A nyitó mérésre egy 46 ítemes papír-ceruza tesztet használtunk. Mivel nem állt rendelkezésre standardizált teszt és próbamérésre nem volt lehetőség, ezért a teszt feladatainak nehézsége nem volt becsülhető. A cél az volt, hogy lehetőleg a vizsgált algoritmikus gondolkodási kompetencia szempontjából ne legyen szignifikáns különbség a kísérleti és kontroll csoportok között.

2. táblázat. A nyitó teszt átlagai és szórásai a kísérleti és kontroll csoportban (százalékpont).

	Statisztikai jellemző	Nyitó teszt
Kontroll csoport	Átlag	64
	Szórás	8
Kísérleti csoport	Átlag	63
	Szórás	11
Teljes minta	Átlag	64
	Szórás	9

A teszt az átlag tanúsága szerint könnyűnek bizonyult (nyitó mérés), viszont a célt sikerült elérni, hiszen a két csoport között nem volt szignifikáns különbség (t-próba:  $F=3,60$ ;  $\text{sig}=0,06$ ;  $t=0,37$   $\text{sig}=0,71$ ).

A záró mérésnél a nyitóteszt megfelelő elkülönítésmutatóval rendelkező itemjeit megtartva (28 ítem), a maradék ítemeket cserélve készítettük el az új mérőeszközt.

### 3.3 Záró mérés

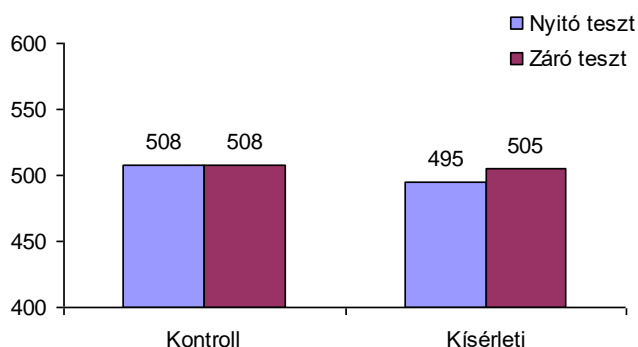
A záró teszt 28 íteme megegyezett a nyitó tesztben szereplő feladatokkal, további 18 új ítemmel kiegészítve.

3. táblázat. A záró teszt átlagai és szórásai a kísérleti és kontroll csoportban (százalékpont).

		Nyitó teszt
Kontroll csoport	Átlag	68
	Szórás	12
Kísérleti csoport	Átlag	66
	Szórás	13
Teljes minta	Átlag	66
	Szórás	13

A záró teszt esetében sem volt szignifikáns a különbség a két csoport között (t-próba:  $F=0,09$ ;  $\text{sig}=0,76$ ;  $t=0,70$ ;  $\text{sig}=0,48$ ).

Ha összehasonlítjuk a nyitó teszt és a záró teszt eredményeit, azt tapasztalhatjuk, hogy az átlagok kis mértékben növekedtek. Mivel a két teszt csak a 28 ítemes magtesztben egyezett meg, de egyik teszten sem volt szignifikáns különbség a két csoport között, ezért standardizálva (átlag: 500; szórás: 100) az eredmények már összehasonlíthatóakká válnak. A standard pontok között a növekedés ugyan számszakilag mindkét csoportnál megjelenik, de egyik esetben sem szignifikáns.



1. ábra. A nyitó és záró teszt átlaga csoportonként standard pontban.

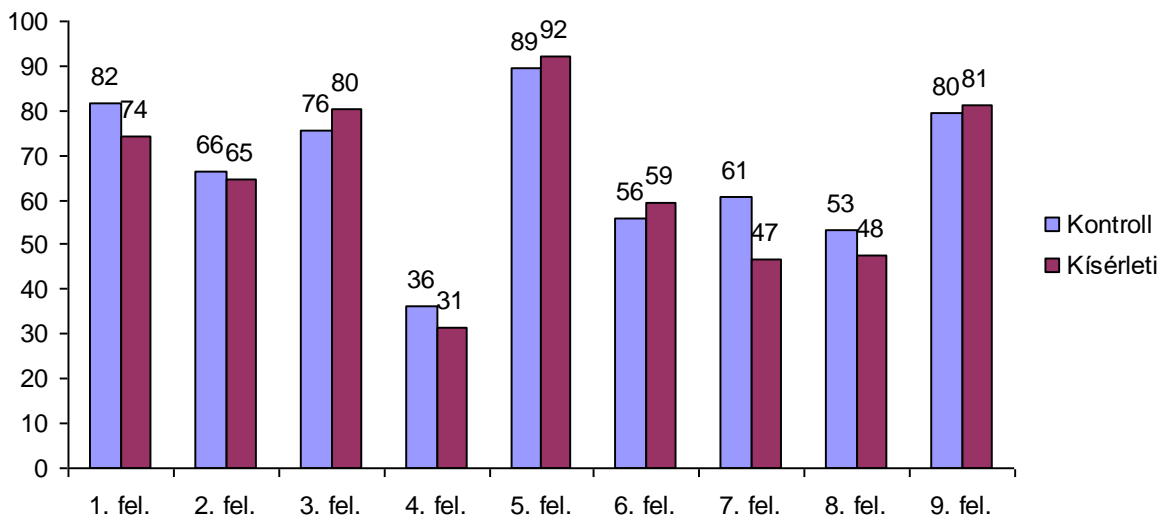
Bár a különbség nem szignifikáns, az mindenképpen figyelemre méltó, hogy a szórás adatok a kísérleti csoport esetében jelentősen csökkentek, míg a kontroll csoport esetében növekedtek. A szórások csökkenése a csoport homogenizálódására utal a vizsgált kompetencia vonatkozásában. A növekvő szórások viszont azt jelzik, hogy a kontroll csoporton (LOGO-Imagine) belül a tanított algoritmus leíró nyelv jobban megosztotta a csoportot. Valószínűsíthető, akik „elveszítették a fonalat” valahol a tanítási folyamat során, azok nehezebben kapcsolódtak be az új témáknál és az ezzel együtt járó motivációvesztés tovább rontotta a tanulási teljesítményt, míg a robotprogramozás esetén ez nem volt ilyen domináns.

A százalékos teljesítmény azonban összességében igen magas, figyelembe véve a tesztek absztrakciós szintjét. Ez mutatja a diákok átlag fölötti gondolkodási képességeit, amit más helyi és országos mérések is jeleznek a kísérleti populáció esetén.

Mivel a kísérleti és kontroll csoport teszteken elért eredményei között nem mutatható ki szignifikáns különbség, ezért érdemes megvizsgálni a feladatonkénti eredményeket, hogy a két csoport tudásában mutatkozik-e eltérés. A két programkörnyezet jelentős eltérése indokolná ezt. Az eredményeket a záró teszt feladatain elemezzük.

4. táblázat. A záró teszt feladatain elért átlagok és szórások a kísérleti és kontroll csoportokban (százalékpont).

	Csoport	Átlag	Szórás
1. feladat	Kontroll	82	14
	Kísérleti	74	16
2. feladat	Kontroll	66	26
	Kísérleti	65	25
3. feladat	Kontroll	76	36
	Kísérleti	80	31
4. feladat	Kontroll	36	32
	Kísérleti	31	36
5. feladat	Kontroll	89	21
	Kísérleti	92	22
6. feladat	Kontroll	56	19
	Kísérleti	59	23
7. feladat	Kontroll	61	39
	Kísérleti	47	40
8. feladat	Kontroll	53	18
	Kísérleti	48	20
9. feladat	Kontroll	80	34
	Kísérleti	81	29



2. ábra. A záró teszt feladatain elért átlagok a kísérleti és kontroll csoportokban (százalékpont).

Az 1. és 7. feladat az, amelyen a kontroll csoport számszakilag jobb átlageredményt ért el, de a különbség nem szignifikáns (1. feladat:  $F=0,06$ ;  $\text{sig}=0,82$ ;  $t=2,05$ ;  $\text{sig}=0,056$ ; 7. feladat:  $F=0,14$ ;  $\text{sig}=0,71$ ;  $t=1,40$ ;  $\text{sig}=0,17$ ).

A feladatok a LOGO nyelvi szintaktikájához illeszkednek az 1. feladat esetén, míg a szokásos tanórai alkalmazásnál megjelenő ábrákhoz a 7. feladat esetén. Mivel a robotprogramozást tanulók esetén ilyen típusú feladatok nem jelentek meg a kísérlet során, így ez a különbség indokolható.

A másik irányú eltérés, tehát a kísérleti csoport javára, a legnagyobb különbség a 3. feladat esetén mutatható ki. Ez sem jelent szignifikáns eltérést ( $F=0,56$ ;  $\text{sig}=0,46$ ;  $t=-0,55$ ;  $\text{sig}=0,59$ ). A feladat tartalma itt a szelektív programszerkezetekre jellemző elágazások értelmezését vizsgálta. A megjelenítés módja az NXT-G programhoz hasonló szerkezetű, de nem grafikus ikonok, hanem szöveges leírás alapján.



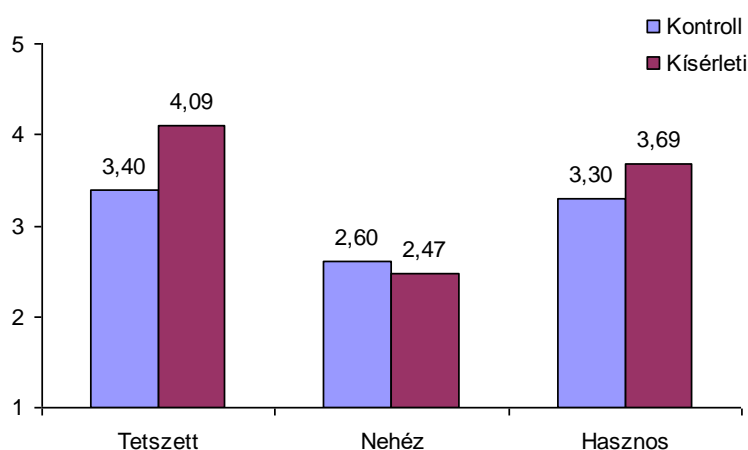
### 3.4 Kérdőív

A kísérlet során az egyetlen szignifikáns különbség a kérdőívvel mért „tetszési index” alapján mutatható ki. A kérdőíven szerepeltek olyan kérdések, amelyek a kísérleti időszakban tanult programozási környezet és módszerek tetszését, nehézségét és hasznosságát vizsgálták a tanulói válaszok alapján.

A kérdések tartalmáról minden esetben egy ötfokú skála értékei közül választva kellett véleményt alkotni.

5. táblázat. A kísérleti időszakban a tanulók válaszai alapján a „tetszési index” mutatószámai (átlag, szórás).

		Tetszett	Nehéz	Hasznos
Kontroll	Átlag	3,40	2,60	3,30
	Szórás	1,04	0,89	0,88
Kísérleti	Átlag	4,09	2,47	3,69
	Szórás	0,73	0,92	1,09
Total	Átlag	3,76	2,53	3,50
	Szórás	0,95	0,90	1,00



3. ábra. A kísérleti időszakban a tanulók válaszai alapján a „tetszési index” mutatószámai (átlag).

A „Mennyire tetszett az informatikaórán tanult programozás?” kérdésre adott válaszok átlagai között szignifikáns különbség mutatkozott a kísérleti csoport javára ( $F=4,22$ ;  $\text{sig}=0,06$ ;  $t=-3,05$ ;  $\text{sig}=0,00$ ). A 4,09-es minősítés a kontroll csoport 3,40-esével szemben azt mutatja, hogy a robotok használata nagymértékben fokozta a motivációt, ami a kompetenciafejlesztés alapja lehet. Ugyanakkor a tanulók (bár nem szignifikáns mértékben), de könnyebbnek és hasznosabbnak is érezték a programozás kísérleti csoportokban használt módon történő megközelítését.

Összegzésül tehát levonhatjuk a következtetést, hogy a kísérlet kezdetén és végén elvégzett papír-ceruza alapú tesztelés során nem adódott szignifikáns különbség egyik csoport javára sem. Az algoritmikus gondolkodást tehát mindkét technika azonos módon fejlesztette. Bár ennek egyértelműsítéséhez nagyobb mintán, részletesebb vizsgálatokra lenne szükség, de a kiindulási alapot a jelen kísérlet képviselheti.

Motivációs hatását tekintve viszont a kísérleti csoport egyértelműen (szignifikánsan) jobb eredményeket mutatott.

A vizsgált tanulói populáció gondolkodási képességeit tekintve átlag felettinek mondható, így valószínűleg ez is hozzájárult, hogy a különbségek nem adódtak szignifikánsnak.

A kísérlet tehát elérte célját. Bebizonyosodott, hogy a választott programozás-oktatási eszköz, a LEGO® cég MINDSTORMS® robotjai alkalmasak a tanórai használatra, az algoritmikus gondolkodás fejlesztésére. A bennük rejlő kreatív lehetőségek a természettudományos oktatás új alapokra helyezését is elősegíthetik. A motivációs szerepük viszont meghaladja a „hagyományos” programozás-oktatásban eddig használt eszközökét.

## 4 Problémamegoldás versenyszinten

Egy oktatási rendszer fontos eleme, hogy ne csak a tömegek számára teremtse meg a lehetőséget az eszközökhöz történő egyenlő esélyű hozzáféréssel, hanem az adott területen különös tehetséget mutatók számára biztosítson olyan színteret, amely további tanulásra motiválhat. A tehetséggondozás egyik szintere a versenyeztetés.

A bemutatott eszközökhöz kapcsolódóan több nemzetközi verseny is tartozik. Ezek összetettsége néha még távolinak tűnik a magyar oktatási gyakorlathoz viszonyítva, de a hazai trendek is ebbe az irányba mutatnak. Egyre több, a komplex tantárgyi kereteken átívelő tudást igénylő csapatverseny indul. Szerencsére a matematika és a természettudomány jár az élen e tekintetben.

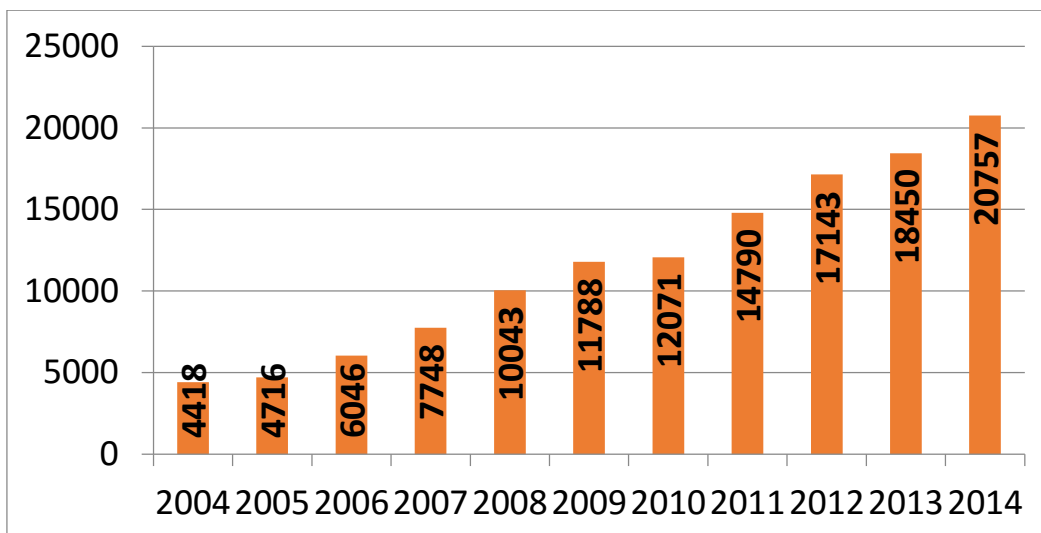
Talán a legnagyobb általános és középiskolai korosztály számára szervezett robotverseny a First Lego League, amely az USA-ból indult 2001-ben és napjainkra az egész világon szerveznek területi fordulókat. Magyarországon 2007 óta rendezik meg 12-16 év közötti diákok, 5-10 fős csapatai számára. A koncepció szerint minden évben egy adott témához kötődően kell egy kutatási projektet lebonyolítani és az eredményeket prezentálni, valamint a MIT mérnökei által tervezett nagyméretű akadálypályán szereplő, problémamegoldás kategóriájába sorolható feladatokat megoldani, saját tervezésű és programozású MINDSTORMS® robotokkal. A versenyen résztvevő csapatokat több szempont szerint értékelik (kutatási prezentáció, csapatmunka, robotfeladatok megoldása, robotdesign). A területi fordulók győztesei juthatnak a következő fordulóba. Világszerte több tízezres a nevezett csapatok száma.

A verseny nem csak a robotika alkalmazását segíti, hanem lehetőséget teremt olyan zömében a természettudományokhoz köthető kutatási projekt lebonyolítására, amelyek nagyon fontosak lennének a magyar oktatási gyakorlatban is.

Példaként egy ilyen projektet említünk. A verseny feladatának apropójaként 2010-ben a „Body forward” versenytémához kapcsolódóan kellett kutatást lebonyolítani. Az orvosbiológia és a mérnöki tudományok összefonódását takaró cím kapcsán a csapatoknak egy általuk választott problémát kellett azonosítani, körüljárni és választ találni a feltett kérdésre. A versenyen a csaptnak által vizsgált kérdés az volt, hogy a jövő orvosi gyakorlatában a kézsérülések területén a műtéti vagy mérnöki technika lesz a hangsúlyosabb. Tehát egy esetleges amputáció során a gyógyítás műtéti technikával (testrész visszavarrás, áthelyezés), vagy mérnöki technikával (művégtag, robotikai eszközök használata) történik-e. Orvosi és kutatómérnöki interjúk során kerestük a kérdésre a választ neves szakemberek segítségével. Az elkészült prezentáció több versenyen is első díjat kapott. A kutatás során összegyűjtött információk olyan plusz motivációt biztosítottak a diákoknak, amely hosszútávon meghatározó volt a pályaválasztás során.

A verseny másik szála a robotika. Itt a világszerte egységes teszt pályán kitűzött feladatok megoldására kell a csapatnak robotot tervezni és építeni. Az egységes feladatspecifikációk igen szűkszavúak. Általános szabály, hogy a feladatleírason és a verseny szabályrendszerén kívül nincs további segítség. A leírás csupán azt tartalmazza, hogy például a pálya adott pontján elhelyezett tárgyat át kell szállítani egy másik adott pontra a robottal. A technikai kivitelezés, amelynek része a robot megépítése és programozása is teljesen a csapatra van bízva. Ezek a feladatok teljes egészében eleget tesznek a szakirodalomban vázolt problémamegoldásnak. Mindez nem tűnhet bonyolultnak, de ha figyelembe vesszük, hogy összesen 15-20 feladat kerül kitűzésre és a robotnak összesen 180 másodperce van valamennyi feladat megoldására, akkor a helyzet már bonyolultabb. Egyetlen robotot lehet használni a megoldás során. A feladatok lényegesen eltérnek egymástól. Kezdve az egyszerű átszállítási feladattól, a bonyolultabb tárgyakból építmény létrehozásáig, vagy a szenzorhasználatot is igénylő felderítési feladattól a tájékozódást igénylő kereséséig. A sikerességhez egy alaprobot megépítése szükséges, amelynél figyelembe vesszük a pálya felépítését a rendelkezésre álló terep adatosságait. A robotra elhelyezett különböző felépítmények segítségével oldhatók meg a feladatok. Mindezek gyors cserélhetősége, modularitása biztosítja a rövid idő alatt a precíz megvalósítást. A versenyekről további információk a <http://www.first-lego-league.org/en/> hivatalos weblapon található.

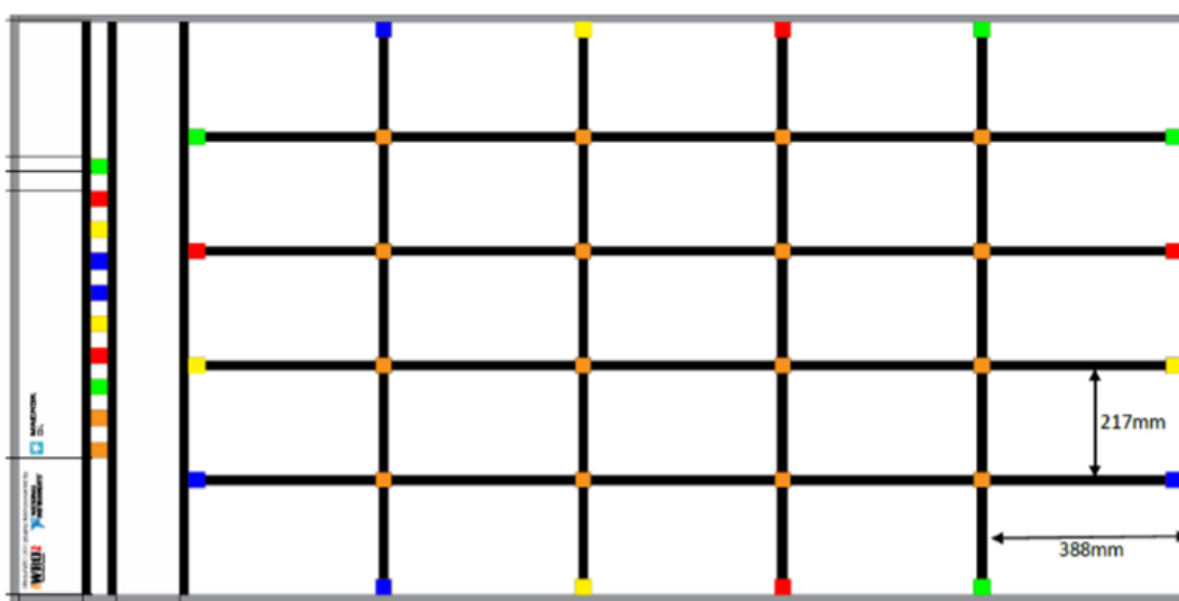
A másik nagy múltú nemzetközi robotika verseny a World Robot Olympiad. Az elmúlt 11 év nevezési statisztikái alapján, 2014-ben közel 21 000 csapat részvételével zajlottak az egyes országok nemzeti döntői.



4. ábra. A WRO nemzetközi verseny részvételi adatainak növekedése (Forrás: <http://www.wrobo.org/>)

A verseny koncepciója lényeges eltér az előzőekben bemutatott First Lego League versenytől. Több kategóriában is lehet indulni, amelyek vagy életkorhatárok, vagy a mögöttes koncepció alapján kerültek meghatározásra. A 2015-ös verseny JHS (Junior High School, 13-16 éves) kategóriájában egy terepasztalhoz kötött robottervezési, építési és programozási feladat megvalósítása volt a cél a három fős csapatoknak.

A feladat lényege, hogy egy nagyméretű négyzetrácsos felületen kellett a robotnak 3,5 cm élhosszúságú különböző színű kockákat összegyűjtenie, amelyek a rácspontokban helyezkedtek el. A négyzetrács egy koordináta rendszert szimbolizált, amelynek a sorait és oszlopait is egy-egy szín azonosította. A pálya bal oldalán található kódsor alapján lehetett a színek kódokat a sorokhoz és oszlopokhoz társítani. Az ábra bal oldalán fentről-lefelé az első négy darab négyzet színe a sorok, míg a második négy négyzet színe az oszlopok azonosítója. Az utolsó két négyzet színe pedig az első kocka sor-oszlop pozíciója. A kódsor a verseny során változott. Az első lépés tehát a robot részéről a kódok beolvasása és az első kockapozíció meghatározása.



5. ábra. A WRO nemzetközi verseny tesztpálya (2015)

A következő kocka pozíciójának sor és oszlop azonosítóját az előző kocka színe és a kocka alatti mező színe kódolta. Tehát a pályát a színek kódok leolvasása és értelmezése alapján kellett

bejárni (kockától kockáig). Nehezítés, hogy a pályán olyan kockák is szerepeltek, amelyekhez nem volt szabad hozzáérni. Tehát a robotnak át kellett haladni felettük. Ezeken kívül korlátozott a használható szenzorok és motorok száma (4 db szenzor és 3 db motor), valamint összesen 180 másodperc állt rendelkezésre a megoldáshoz, és az indítás után már nem lehetett a robothoz hozzáérni.

A feladat megértése után a robot tervezése, megépítése és programozása a csapat feladata. Ha megértettük a feladatot, akkor egy laikus számára nem tűnik bonyolultnak, de a részletekben történő elmélyedés után jelentkeznek a problémák. Ezek közül néhány a teljesség igénye nélkül:

- Ahhoz, hogy a robot áthaladjon a tiltott kockák felett magasra kell helyezni a színérzékelő szenzorait, amelyeket így a környezet fénye zavar.
- Egyetlen motor használható a kocka felvételére és robotra helyezésére, hiszen a mozgáshoz két motor szükséges.
- A pályát neon világítja meg, amelynek vibrálási frekvenciája a szenzor mintavételei frekvenciatartományába esik.
- A pályán való tájékozódáshoz nem csak a kiindulási és célkoordináták ismerete szükséges, hanem a robot aktuális haladási iránya is, hiszen ez fogja meghatározni a fordulások irányát.
- A cél meghatározáshoz a kocka színének beolvasása mellett a kocka alatti mező színének leolvasása is szükséges. Ezek távolsága legalább 3,5 cm.
- A pontos pozíció meghatározásához érzékelni kell a koordináta rendszer vonalait, ugyanakkor az egyenes mozgáshoz és a fordulásokhoz a vonalak követése szükséges.

A vázolt problémák csak ízelítők abból, hogy milyen bonyolultságú feladatot kell megoldani a 13-16 éves diákoknak. Ezek egy része a konstrukció kialakításával, míg más része programozás-technikai eszközökkel oldható meg.

A nemzetközi versenyek jól példázzák, hogy van lehetőség az általános és középiskolás diákok számára a komoly műszaki és programozási kompetenciák elsajátítására. Az informatika nem csak a felhasználói szoftverhasználatról szólhat. Mindez Magyarországon is egyre nagyobb teret hódít és segíti a felsőoktatási képzésbe bejutott tanulók és oktatók munkáját.

## 5 Összegzés

A tanulmányban gyakorlati szempontból mutattunk be egy olyan eszközt, amellyel az informatika oktatás algoritmikus gondolkodás fejlesztési részét lehet akár kisiskolás kortól hatékonyan és tanulói motivációt erősítően fejleszteni.

A tapasztalatok és a kismintás pedagógiai kísérlet meggyőző lehet a felhasználás jogosságát illetően. Az alkalmazás gátló tényezői közül a pénzügyi érvek mellett a tanári innováció hiánya, amit legfontosabbnak tartunk.

Az eszköz használatával olyan kompetenciák, képességek fejleszthetők, amely más tantárgyak, tudományterületek eszközrendszerét és bővítik és hatással vannak rájuk. Az analízis, rendezés, absztrakció, összehasonlítás, szintézis, általánosítás, konkretizálás, analógia, algoritmikus gondolkodás, problémamegoldás, konstrukciós készség, természettudományos mérési-elemzési képességek fejlődésével olyan komplex gondolkodás és szemlélet alakítható ki, amely megalapozhatja az egyre inkább preferált műszaki- és természettudományos továbbtanulást, gondolkodásmódot.

Az oktatáspolitikai előtt tehát lehetőségként ott áll egy koncepció, amely támogatásával az informatika oktatás megújítható.

## Felhasznált források

- [1] Nagy József (2000): XXI. század és nevelés. Osiris Kiadó, Budapest
- [2] Szántó Sándor (2002): Az algoritmikus gondolkodás fejlesztése általános iskolában Új pedagógiai szemle, 2002/5: 84-175.
- [3] Pentelényi Pál (1999): Az algoritmikus szemléletmód kialakítása és fejlesztése a tanítástanulási folyamatban. LIGATURA Kiadó, Budapest
- [4] B. Erwin (2001): Creative Projects with LEGO Mindstorms, Addison-Wesley Professional

- [5] Bessenyei István (1998): Képernyő, tanulási környezet, olvasás: Seymour Papert tanuláseméleti nézeteiről az olvasás kapcsán. Új Pedagógiai Szemle, 1998/10. sz.
- [6] Csapó Benő (2003): A képességek fejlődése és iskolai fejlesztése. Akadémiai Kiadó, Budapest
- [7] Lénárd Ferenc (1978): A problémamegoldó gondolkodás, Budapest, Akadémiai kiadó
- [8] M. Gasperi, P. „Philo” Hurbain, I. Hurbain (2007): Extreme NXT, Extended the LEGO Mindstorms NXT to the Next Level, Apress, Inc.
- [9] D. Astolfo, S. Cavers, K. Clague, Cs. Soh, L. Whitman, T. Witherspoon (2002): LEGO Mindstorms Ultimate Builder Projects, Syngress Publishing, Inc.
- [10] Kiss Róbert, Pásztor Attila (2009): Mobil robotok programozása NXC és NXT-G nyelven. Kecskeméti Főiskola GAMF Kar, főiskolai jegyzet.
- [11] Kiss Róbert, Badó Zsolt (2011): Egyszerű robotika, a Mindstorms NXT robotok programozásának alapjai, National Instruments Hungary, digitális könyv (<http://www.amcham.hu/other-publications>)
- [12] Kiss Róbert (2014): Kreatív robotika, a MINDSTORMS EV3 robotok programozásának alapjai, National Instruments Hungary, Digitális könyv ([http://www.hdidakt.hu/adat/dw\\_anyagok/dw\\_74.pdf](http://www.hdidakt.hu/adat/dw_anyagok/dw_74.pdf))