ساختن‌گرایی

نویسنده: یاسمین کافای[1]
ترجمه و تألیف: خسرو داودی
مؤلف کتاب‌های درسی ریاضی

اشاره

پیش از مطالعه متن، توجه خوانندگان عزیز را به نکات زیر جلب می‌کنم:

۱. متن حاضر، بخشی از فصل سوم کتاب «راهنمای علوم یادگیری»[2] است که توسط «دانشگاه کمبریج» در سال ۲۰۰۶ منتشر شد. سرویراستار این مجموعه، کیت سایر[3] است. با توجه به طولانی بودن متن، قسمتی از متن ترجمه و تقدیم می‌شود. ابتدا قصد داشتم همهٔ این فصل را در سه شماره و به‌تدریج ارائه کنم، اما با نظر اعضای محترم هیئت تحریریه، نکات اصلی در یک متن انتخاب و ترجمه شدند. همچنین به پیشنهاد دوستان توضیحاتی تکمیلی توسط مترجم به متن اصلی اضافه شد تا برای خواننده روشنگر و مکمل متن اصلی باشد. این قسمت‌ها با رنگ متفاوت درج شده‌اند.

۲. در ترجمهٔ این متن برای واژهٔ «constructionism» که یک لغت ترکیبی و ابداعی است و به‌طور معمول در واژه‌نامه‌های انگلیسی نیز وجود ندارد، از معادل «ساختن‌گرایی» استفاده شده است. شباهت این عبارت با واژهٔ آشنای «ساخت‌گرایی» در علوم یادگیری ممکن است ابهام ایجاد کند. مطالعهٔ متن به رفع این ابهام کمک می‌کند. همچنین اصطلاح «ساخت و سازگرایی» به‌عنوان معادل «constructivism» از بین ترجمه‌های متفاوتی که برای این واژه در فارسی به‌کار رفته، انتخاب شده است.

۳. مفهوم ساخت و سازگرایی در ادبیات پژوهشی و دانشگاهی ایران شناخته شده است. مقالات، کتاب‌ها و مطالب متفاوتی به زبان فارسی در این باب منتشر شده‌اند، اما مفهوم ساختن‌گرایی کمتر مورد توجه بوده است. اغلب آموزشگران ریاضی یا از آن اطلاع کمی دارند و یا تفاوت آن را با ساخت و سازگرایی به روشنی نمی‌دانند. در طول سال‌های انتشار مجلهٔ «رشد آموزش ریاضی»، تنها در شمارهٔ ۷۵ در سال ۱۳۸۳ و در حد یک‌ونیم صفحه به این موضوع پرداخته شده و معرفی مختصری از مفهوم ساختن‌گرایی، با ترجمهٔ آقای مرتضی ایوبیان درج شده است. خوش‌بختانه ایشان نیز از همان معادل‌های به کار رفته در متن‌ها و ترجمهٔ خود استفاده کرده‌اند.

۴. با گسترش حضور رایانه‌ها در مدرسه‌های ایران در دههٔ ۱۳۷۰، بحث‌های زیادی در خصوص محتوای قابل ارائه به دانش‌آموزان مطرح شد. در آن سال‌ها عده‌ای زبان برنامه‌نویسی «لوگو» را با توجه به سادگی آن، نقطهٔ شروع مناسبی برای آموزش زبان‌های برنامه‌نویسی و تفکر الگوریتمی معرفی کردند و تعدادی از مدرسه‌ها نیز آن را در برنامهٔ آموزش رایانهٔ خود گنجاندند. اما اغلب معلمان و آموزشگران از وجود نرم‌افزار لوگو که در واقع برای آموزش ریاضی طراحی شده بود، بی‌اطلاع بودند و بعد از مدتی نیز لوگو فراموش شد. در این متن با فلسفه و چگونگی به‌وجود آمدن این نرم‌افزار در آموزش ریاضی آشنا می‌شویم.

۵. با توجه به تغییرات گستردهٔ کتاب‌های درسی و تأکید مؤلفان در استفاده از فعالیت‌های آموزشی و روش‌های اکتشافی که مبتنی بر نظریه‌های ساخت‌وسازگرایی هستند، عموم معلمان به نوعی با فعالیت‌هایی که به ساختن دانش توسط یادگیرنده‌ها کمک می‌کند، آشنا شده‌اند. لذا مطالعه این متن می‌تواند به درک بهتر تفاوت ساخت‌وسازگرایی و ساختن‌گرایی کمک کند و افق تازه‌ای را برای معلمان و دبیران ایجاد نماید.

کلیدواژه‌ها: ساختن گرایی، ساخت و سازگرایی، علوم یادگیری، آموزش، استفاده از رایانه در آموزش، زبان برنامه‌نویسی لوگو، شبیه‌سازها، بسته‌های دست‌ورزی

نقطهٔ اشتراک پژوهشگران علوم یادگیری در این است که خود را متعهد می‌دانند، از یادگیری انتقالی به سبک انتقال و اکتساب که با روش‌های سخنرانی و آزمون‌های پی‌درپی آمیخته است، بپرهیزند و به سمت روش‌های فعال و مشارکتی حرکت کنند. شاید اولین آموزشگری که پی برد رایانه‌ها چنین فرصتی را برای مدرسه‌ها ایجاد می‌کنند، **سیمور پَپرت**[۴] بود؛ همان شخصی که زبان معروف و شناخته‌شدهٔ برنامه‌نویسی «لوگو»[۵] را خلق کرد. پَپرت دو دکترا در ریاضیات دریافت کرد، اما توسعهٔ حرفه‌ای خود را در مطالعات شناختی زیر نظر **ژان پیاژه**[۶] بنیانگذار ساخت و سازگرایی ـ یکی از تئوری‌های بنیادین علوم یادگیری امروز ـ گذارند. پَپرت بعد از ترک آزمایشگاه پیاژه در سوئیس، یک کرسی در انیستیتو تکنولوژی ماساچوست گرفت؛ جایی که او با همکاری **ماروین مینسکی**[۷] آزمایشگاهی در زمینهٔ هوش مصنوعی پایه‌گذاری کرد. در سال ۱۹۷۰، پَپرت گسترش دیدگاه شناختی ساخت و سازگرایی پیاژه را با تبیین اصول پداگوژیک آن آغاز کرد که بعدها این چارچوب تأثیر زیادی بر پژوهشگران علوم یادگیری گذاشت.

ساختن‌گرایی، ساخت و سازگرایی نیست، همان‌طور که نه پیاژه هرگز قصد داشت نظریه‌اش را در بارهٔ دانش به‌عنوان نظریهٔ یادگیری و تدریس مطرح کند و نه یادگیری ساختن‌گرا ساده‌شدهٔ یادگیری اکتشافی است

وقتی در سال ۱۹۸۰ کتاب پَپرت با عنوان «توفان ذهن»[۸] منتشر شد، هنوز واژهٔ ساختن‌گرایی درست نشده بود. در این کتاب و آثار پس از آن، او تئوری خود را برای یادگیری، تدریس و طراحی توسعه داد. بسیاری از افراد، ایدهٔ «بچه‌ها، رایانه و ایده‌های قدرتمند» که زیر عنوان کتاب پپرت بود، به‌عنوان نسخهٔ ساده شدهٔ یادگیری اکتشافی پیاژه همراه با زبان برنامه‌نویسی لوگو تعبیر کردند که در واقع این فکر و ایده را درست در نقطهٔ مقابل کاربردش قرار می‌داد. **ساختن‌گرایی، ساخت و سازگرایی نیست، همان‌طور که نه پیاژه هرگز قصد داشت نظریه‌اش را دربارهٔ دانش به‌عنوان نظریهٔ یادگیری و تدریس مطرح کند و نه یادگیری ساختن‌گرا ساده‌شدهٔ یادگیری اکتشافی است.** بنابراین در نقطهٔ مقابل هر نوع «تدریس»[۹] (آموزش تجویزی و دستورالعملی) است. در نهایت در ساختن‌گرایی، کاربردها و نه رایانه‌ها به‌عنوان راهبران تغییر در آموزش دیده می‌شوند. دیدگاه‌های یادگیری ساختن‌گرایی پَپرت، به معنی ساختن رابطه‌ها بین دانش قدیم و جدید در تعامل با دیگران، در حالی که مصنوعات مرتبط با جمع را می‌سازند، شکل می‌گیرد. بنابراین هر فصلی دربارهٔ ساختن‌گرایی با روشن کردن سه نظریه آغاز شود: ساخت و سازگرایی، تدریس (آموزش تجویزی و دستورالعملی) و آموزش فناورانه‌محور. قبل از آنکه به واکاوی‌های خاص نظری و پداگوژیک بپردازیم، یادآور می‌شویم که **شباهت‌های نزدیک ساختن‌گرایی و ساخت و سازگرایی پیاژه غالباً باعث بدفهمی می‌شود، در حالی که هنوز تفاوت‌های آشکاری بین این دو وجود دارد:**

ساختن‌گرایی معنای ضمنی ساخت و سازگرایی را برای یادگیری، به‌عنوان ساختن ساختارهای دانش، بدون توجه به شرایط یادگیری به اشتراک می‌گذارد. سپس این ایده را اضافه می‌کند که این شرایط به‌طور مقتضی در زمینه‌ای که یادگیرنده آگاهانه درگیر ساختن موجودی عام است، اتفاق می‌افتد که می‌تواند برای مثال یک قلعهٔ شنی در ساحل دریا و یا یک نظریه دربارهٔ جهان باشد [پَپرت، ۱۹۹۱:۱].

ساختن‌گرایی همیشه وفاداری‌اش را به تئوری پیاژه تصدیق می‌کند، اما قطعاً با آن یکسان نیست. در حالی که ساخت و سازگرایی در زمینهٔ توسعهٔ فردی و ساختارهای مجزای دانش برتری دارد، ساختن‌گرایی بر رابطه‌های طبیعی و معمول دانش‌ها با ابعاد شخصی و اجتماعی تمرکز دارد. این ترکیب جنبه‌های فردی و اجتماعی در یادگیری، در قلب بسیاری از مباحث علوم یادگیری قرار دارد. تقابل ساختن‌گرایی و آموزش‌های مستقیم و دستورالعملی، غالباً یادگیری ساختن‌گرا را با یادگیری اکتشافی از نظر یادگیری بدون برنامهٔ درسی که در آن، بچه‌ها اصول و ایده‌های خودشان را کشف می‌کنند، هم‌سو می‌کند. یک وجه مشترک دیگر با ساخت و سازگرایی این ایده است که هر تدریس انتقالی و دستورالعملی برای آموزش بد است. یک مطلب خواندنی به روزتر از نوشته‌های اصیل پَپرت این نظر را بیشتر آشکار می‌کند:

اما تدریس بدون برنامهٔ درسی معنایش بدون اراده بودن و بدون شکل بودن کلاس یا به‌طور ساده‌تر، **تنها رها کردن بچه‌ها** نیست؛ معنایش حمایت کردن از بچه‌هاست، به‌طوری که آن‌ها ساختارهای هوشمندانهٔ خودشان را با مواد در اختیارشان براساس فرهنگ پیرامونی خود بسازند. در این مدل، مداخله‌های آموزشی به معنی تغییر فرهنگ‌ها، طرح‌ریزی عناصر ساختنی جدید در آن و همچنین زدودن سم‌ها است [پَپرت، ۱۹۹۳-۱۹۸۰: ۳۱]

ساختن‌گرایی، آن بخش از دیدگاه تدریس و آموزش را که بیشتر شناخته‌شده‌اند، به خوبی تبیین می‌کند. یادگیری و تدریس در تعامل بین معلم و شاگردان زمانی ساخته می‌شود که آن‌ها درگیر طراحی، یادگیری و بحث دربارهٔ دست‌سازه‌ها می‌شوند. علاوه بر آن، چنین تعامل‌های یادگیری تنها به مدرسه‌ها محدود نیستند، بلکه به گروه‌های کوچک یادگیری و خانواده‌ها بسط و توسعه داده می‌شوند. نحوهٔ طراحی کردن محیط‌های یادگیری که تسهیل کنندهٔ هم‌افزایی و اشتراک ایده‌ها باشد، نقطهٔ تمرکز بسیاری از تلاش‌ها در علوم یادگیری است.

پَپرت در نوشته‌های خود اشاره می‌کند که روزی در دانشگاه از کنار کلاس هنر عبور می‌کرد و شاهد بود که چطور دانشجویان

بـــا برش زدن به یک قالب صابون ســـعی می‌کردنـــد، فرم‌ها و شکل‌های متفاوت بسازند. او از آن لحظه فکر می‌کرد که چطور می‌شود در درس ریاضی نیز چنین محیطی را فراهم کرد. ایدهٔ توسعهٔ نرم‌افزار لوگو از آنجا آغاز شد.

زبان برنامه‌نویس لوگو همیشه همراه و در کنار ساختن‌گرایی بوده اســـت. این موضوع بسیاری را به این باور که ساختن‌گرایی فناوری را به‌عنوان نیروی پیشران برای نحوهٔ آموزش و یادگیری می‌بینند، رهنمون ساخته است. هنوز همان‌طور که پَپرت استدلال می‌کند، این نوع از تفکر فناورانه‌محور مهم‌تر و شایسته‌تری به‌نظر می‌رسد، از اینکه فناوری را عامل تغییر آموزش بدانیم:

آیا می‌توان با چوب خانه‌های خوب ساخت؟ اگر من خانه‌ای با چوب بسازم و خراب شود، آیا نشان‌دهندهٔ این است که با چوب نمی‌توان خانه‌های خوبی ســـاخت؟ آیا چکش‌ها و اره‌ها مبلمان خوب می‌سازند؟ این سؤال‌ها خودشـــان را به‌عنوان سؤال‌های فناورانه‌محور بـــا انکار نقش افراد و عناصر زیر ســـؤال می‌برند. مهارت‌ها، طراحی و زیبایی‌شناسی صرفاً ساختهٔ دست افراد است [پَپرت ۱۹۸۷: ۲۴].

ســـاختن‌گرایی ما را به چالش می‌کشـــد تـــا در افکارمان درخصوص یادگیری و تدریس تجدیدنظر کنیم. برنامه‌نویسی با لوگو بستری را برای آزمایش کردن فراهم می‌کند تا دانش‌آموزان را درگیر حل مســـئله و یادگیری چگونه یاد گرفتن کند. علاوه بر آن، برنامه نویســـی با لوگو راه‌های متفاوت یادگیری مفهومی ریاضیات و علوم با اســـتفاده از رایانه را نشان می‌دهد. بسیاری از ایـــن چالش‌ها در یادگیری و تدریس در رابطه با علوم یادگیری ادامه خواهند داشت، خواه درگیر استفاده از رایانه باشند یا نباشد.

پَپرت در دست‌نوشته‌هایش به خاطرات خود از زمان کودکی اشاره می‌کند که چطور با استفاده از اتصال چرخ‌دنده‌های کوچک به یکدیگر، قوانین مهم فیزیکی را تجربه کرده اســـت. او امیدوار اســـت دست‌ورزی با رایانه چنین محیطی را برای کشف و درک بهتر مفاهیم ریاضی فراهم کند.

هدف مقالهٔ حاضر این اســـت که دیدگاه‌های ساختن‌گراها را با وضوح بیشـــتری توضیح دهد تا ماهیت دانستن، تدریس و یادگیری تبیین شود. در ادامه ریشه‌های تاریخی ساختن‌گرایی و اســـتفاده از لوگو به‌عنوان مثال بیان شده است و بحث با ایدهٔ کلیدی ساختن‌گراها دربارهٔ ساختن دانش، فرهنگ‌های یادگیری و کاربردهای ســـاختن دانش، به ســـمت طراحی با محیط‌های شبیه‌سازی و بســـته های دست‌سازه سوق می‌یابد. سپس یک مطالعهٔ مـــوردی از یادگیری با نرم‌افزار و طراحی فعالیت‌ها ارائه شـــده است تا کاربســـت ایده‌های مرکزی ساختن‌گراها با مثال روشن و واضح شود. در قسمت نتیجه‌گیری، نظریات ماندگار و چالش‌ها در خصوص ســـاختن‌گرایی و در حوزهٔ علوم یادگیری مورد بحث قرار می‌گیرد.

ریشه‌های تاریخی

در تمام موارد مرتبط با ساختن‌گرایی، زبان برنامه‌نویسی لوگو، به علت در برداشتن نظراتی که باعث تغذیه کردن مباحث مربوط به استفاده از رایانه در مدرســـه‌ها در اوایل دههٔ ۱۹۸۰ می‌شد، «موضوعی مهیج»[10] ـ عبارتی که به نام **شری ترکل**[11] ثبت شده اســـت ـ به شمار می‌رفت و حضوری پررنگ داشت. در آن زمان رایانه‌ها آمادهٔ خروج از آزمایشگاه‌های دانشگاه‌ها و حضور در دنیا بودند، اما کار با رایانه به‌عنوان حیطهٔ اختصاصی بزرگ‌سالان دیده می‌شد. لوگو اولین زبان برنامه‌نویسی نبود که بچه‌ها از آن استفاده می‌کردند. «بیسیک»[12] در بسیاری از مدرسه‌ها برجسته بود و در واقع بحث‌های قابل‌توجهی در مورد اینکه چه زبان برنامه‌نویسی برای مدرســـه‌ها مناسب‌تر است، مطرح بودند. اما در مقابل زبان بیسیک، یادگیری با لوگو متعهد بود که چیزی فراتر از یادگرفتن یک زبان برنامه‌نویســـی ارائه کند که شامل یادگیری نحوهٔ فکر کردن خود فرد و یادگرفتن ریاضی و علوم از مســـیری مفهومی و جدید باشـــد. همین خصوصیات مکمل و اضافه‌شده، لوگو را بی‌شباهت با هر زبان دیگر برنامه‌نویسی ساخت.

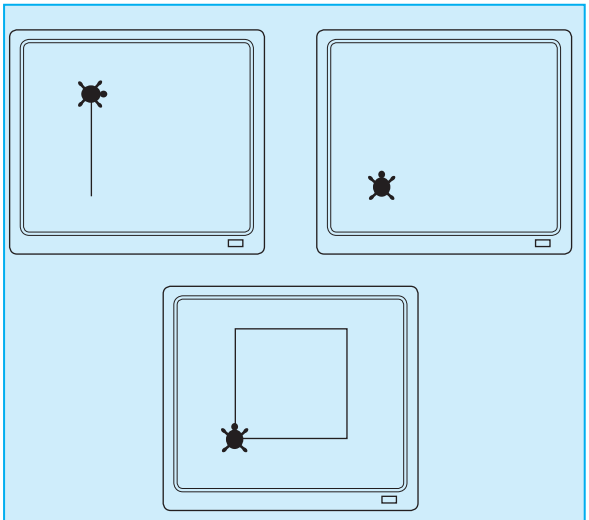
در حالی که ساخت و سازگرایی در زمینهٔ توسعهٔ فردی و ســـاختارهای مجزای دانش برتری دارد، ســـاختن‌گرایی بر رابطه‌های طبیعی و معمول دانش‌ها با ابعاد شخصی و اجتماعی تمرکز دارد

نرم‌افزار لوگـــو را به‌راحتـــی می‌توانید از طریـــق اینترنت دانلـــود کنیـــد. البتـــه توجـــه داشـــته باشـــید کـــه در حال حاضر نســـخه‌های بســـیار زیادی از ایـــن نرم‌افـــزار (حتی با نام‌هـــای مختلـــف) وجـــود دارد. توصیه می‌کنم برای شـــروع MSW LOGO را که نسخهٔ مقدماتی و سادهٔ آن است، انتخاب کنید. پس از نصب و اجرای نرم‌افزار دو بخش اصلی مشـــاهده می‌کنید: یک صفحهٔ نمایش که در وسط آن یک لاک‌پشتی قرار دارد (در نسخه‌های ساده به‌جای لاک‌پشت یک مثلث می‌بینید) و یک قســـمت برای نوشتن دستورات. دســـتورها و فرمان‌های اجرایی به زبان انگلیســـی‌اند و اغلب به صورت مخفف هستند. برای مثال «FD100» یعنی به انـــدازه ۱۰۰ واحد برو جلو. FD مخفـــف کلمهٔ Forward اســـت. در ادامهٔ مطلب با کاربرد چند دستور آشنا می شوید.

اولین ویژگی ذکرشـــده دربارهٔ لوگو این اســـت که چگونه یادگیرنده با لوگو تعامل می‌کرد. بچه ها دستوراتی را می‌نوشتند تا یک شیء گرافیکی ـ که لاک‌پشت نامیده می‌شد ـ را در صفحهٔ نمایش حرکت دهند؛ بیشتر از آنکه با آرایه‌های عددی یا نمادها دست‌ورزی و پردازش کنند. (شکل ۱، سمت راست را ببینید). برنامه‌نویسی رایانه‌ای هم‌معنی با برنامه دادن به لاک‌پشت بود. یک برنامه‌نویس فرمان‌ها و دستوراتی به لاک‌پشت می‌دهد؛ مثل

لاک‌پشت لوگو، به‌عنوان اولین نمایندهٔ ریاضیات رسمی، به بچه‌ها کمک می‌کرد و آن‌ها را قادر می‌ساخت با مجسم کردن بدن خود بتوانند چگونگی حرکت لاک‌پشت روی صفحهٔ نمایش را تصور کنند

«۱۰ قدم جلو برو» و سپس «۹۰ درجه به راست بچرخ» که در زبان برنامه‌نویسی لوگو با عبارت‌های«FORWARD۱۰» و «RIGHT۹۰» نوشته می‌شود. (شکل ۱، سمت چپ را ببینید). در نتیجه لاک‌پشت روی صفحهٔ نمایش حرکت و بنابراین بازخوردهای تصویری روی آن فراهم می‌کرد؛ چه برنامهٔ نوشته‌شده درست بود یا نبود. به‌علاوه، لاک‌پشت قلمی با خود حمل می‌کرد که می‌توانست رسم کند و اثری از قدم‌هایش روی صفحه باقی بگذارد. دستورهای: «قلم را روی صفحه بگذار[۱۳]»، «۱۰ قدم جلو برو» و «۹۰ درجه به راست بچرخ»، در صورتی که چهار بار تکرار می‌شدند، به رسم یک مربع روی صفحهٔ رایانه منجر می‌گردیدند. (شکل ۱، قسمت وسط را ببینید).



شکل ۱ـ تصویر سمت راست یک لاک‌پشت را در موقعیت شروع که آمادهٔ دریافت فرمان‌هاست نشان می‌دهد. در تصویر سمت چپ، لاک‌پشت فرمان‌های «قلم را روی صفحه بگذار»، «۱۰ قدم جلو برو» و «۹۰ درجه به راست بچرخ» را اجرا کرده است. تصویر قسمت وسط، رسم‌شدهٔ یک مربع را با ۴ بار تکرار دستورات یا [REPEAT4 [FORWARD 100, RIGHT90، نشان می‌دهد.

دومین ویژگی قابل توجه این است که لاک‌پشت لوگو، به‌عنوان اولین نمایندهٔ ریاضیات رسمی، به بچه‌ها کمک می‌کرد و آن‌ها را قادر می‌ساخت با مجسم کردن بدن خود بتوانند چگونگی حرکت لاک‌پشت روی صفحهٔ نمایش را تصور کنند. به دستورها و فرمان‌های زیر توجه کنید که چگونه لاک‌پشت یک قدم به جلو حرکت می‌کند، یک درجه به راست می‌چرخد، و سپس ۳۶۰ مرتبه این دستورات را تکرار می‌کند:

REPEAT 360 [FORWARD 1, RIGHT 1]. با گذاشتن قلم روی صفحه، این دستورها یک دایره روی صفحهٔ نمایش رسم می‌کند. پَپرت این ویژگی را «یادگیری همانندساز[۱۴]» نامید و بسیار مهم توصیف کرد، چون به بچه‌ها اجازه می‌داد با اشیای رایانه‌ای به روش‌های متفاوت فرا گیرند:

برای مثال، دایرهٔ لاک‌پشتی، همانندسازی‌شدهٔ حرکت بدن است چرا که این دایره ارتباط زیادی با حس و فهم بچه‌ها از چگونگی حرکت قدم‌های خودشان دارد. یا اینکه نوعی همانندسازی با خود است که ارتباط معناداری با حس بچه‌ها از خودشان دارد؛ به عنوان کسانی که اراده، هدف، مطلوبیت، دوست داشتن و نداشتن و ... دارند. هر کس می‌تواند این موضوع را به‌عنوان فرهنگ همانندسازی ببیند که وقتی یک دایره رسم می‌شود، لاک‌پشت با ایده‌هایی از زاویه و ایدهٔ راندن و حرکت دادن مرتبط می‌شود که به‌طور عمیقی ریشه در تجربه‌های بچه‌ها از کارهای فوق‌برنامه‌ای ایشان دارد. [۱۹۹۳/۱۹۸۰، ص ۶۸ - ۶۳]

لاک‌پشت لوگو به بچه‌ها اجازه می‌داد تا با اشیای روی صفحهٔ نمایش دست‌ورزی کنند، همان‌طور که می‌توانستند در دنیای فیزیکی با آن‌ها دست‌ورزی کنند. بنابراین، هندسهٔ لاک‌پشتی یک ورودی ملموس به دنیای رسمی ریاضیات فراهم می‌کرد و به یادگیرنده‌ها اجازه می‌داد که تجربیات شخصی خود را با مفاهیم و عملیات ریاضی مرتبط کنند.

ویژگی ذکرشده در بالا یکی از اساسی‌ترین نکات در تئوری ساختن‌گرایی است. همهٔ ما معتقدیم اگر بتوانیم مفهومی را به شخص دیگری آموزش دهیم، به این معناست که آن موضوع را خودمان خوب درک کرده و فهمیده‌ایم. کاربر لوگو در واقع باید به لاک‌پشت آموزش دهد. این کار زمانی محقق می‌شود که کاربر خود را جای لاک‌پشت تصور کند. برای انجام یک حرکت، مسیر را با بدن خود طی کند و انجام دهد، سپس آن را با نوشتن دستورات به لاک‌پشت فرمان دهد. برای مثال اگر قرار است در صفحه با حرکت لاک‌پشت یک دایره ایجاد شود، او باید با حرکت دادن بدن خود روی یک مسیر دایره‌ای چگونگی حرکت را تصور نماید و سپس با نوشتن دستورهای مورد نظر، آن حرکت را برای لاک‌پشت شبیه‌سازی کند. در واقع اگر بخواهیم روی یک مسیر دایره‌ای حرکت کنیم، ابتدا یک قدم بسیار کوچک برمی‌داریم. سپس با یک زاویهٔ بسیار کوچک می‌چرخیم و دوباره آن را تکرار می‌کنیم. ترجمهٔ این حرکت بدن در زبان لوگو می‌شود (۱ واحد برو جلو، ۱ درجه بچرخ) و این کار را آن قدر ادامه بده تا دایره رسم شود.

سومین ویژگی مهم برنامه‌نویسی لوگو این ایده بود که بچه‌ها دربارهٔ چگونه فکر کردن و یادگیری خود، یاد می‌گرفتند، که بازتاب یا «فراشناخت» نامیده می‌شد. پَپرت مدعی شد که در یادگیری برنامه‌نویسی، بچه‌ها یاد می‌گیرند که دستورات و

فرمان‌ها را بازگو کنند، تکرارها را تشخیص دهند و فکرهای خود را وقتی که برنامه‌ها مطابق انتظارشان کار نمی‌کنند، غلط‌گیری و تصحیح کنند. «اما فکر کردن دربارهٔ یادگیری با قیاس کردن برنامهٔ نوشته‌شده و خروجی آن، یک راه قدرتمند و قابل دسترس برای شروع تبدیل شدن به بازتاب کنندهٔ راهبردهای تصحیح و غور کردن بیشتر برای بهبود و ارتقای تفکر خود است» (۱۹۸۰/۱۹۹۳، ص ۲۳). برنامه‌نویسی با رایانه می‌تواند به «موضوعی برای تفکر با» تبدیل شود که به بچه‌ها کمک کند تا بر عملکرد خود از طریق راه‌های مشابه تجربه‌های یادگیرنده‌ها بازتاب داشته باشند.

بنابراین لوگو هدف‌های چندگانه‌ای را ترکیب کرد: یاد گرفتن برنامه‌نویسی، یادگیری ریاضیات، و یادگیری چگونه یاد گرفتن. این ادعاها بدون معارض نبودند. خیلی‌ها دربارهٔ موفقیت‌ها و

زبان برنامه‌نویس لوگو همیشه همراه و در کنار ساختن‌گرایی بوده است. این موضوع بسیاری را به این باور که ساختن‌گرایی فناوری را به‌عنوان نیروی پیشران برای نحوهٔ آموزش و یادگیری می‌بینند، رهنمون ساخته است

شکست‌های استفاده از لوگو در مدرسه مطالبی نوشته‌اند که پشتوانهٔ ارزشمندی برای گفت‌وگو در این زمینه فراهم کرده است. در تحلیلی فوق‌العاده که از زمینه‌های تاریخی این موضوع در آمریکا و اروپا انجام شده است، **ریچارد ناس**[۱۵] و **سلیا هویلز**[۱۶] (۱۹۹۶) چند مورد مهم‌تر از تأثیرات قوی فرهنگی لوگو را در بازی تشخیص دادند که به منتقدان اجازه می‌داد سؤال‌هایی مشخصاً در مورد لوگو و نه سایر برنامه‌ها مطرح کنند. در بسیاری از مدرسه‌ها، سؤال‌هایی در مورد سودمندی یادگیری با لوگو به‌طور خاص بر انتقال مهارت‌های حل مسئله متمرکز شده بودند و کمتر به فایده‌های یادگیری ریاضیات و ایده‌های اصلاحی و بهبود توجه داشتند. تعداد کمتری از مطالعات، از جمله مطالعات **روی پی**[۱۷] و **میدین کورلانـد**[۱۸] (۱۹۸۴)، به نتایجی مثل یادگیری برنامه‌نویسی با لوگو محصول خاصی تولید نمی‌کند و تأثیرگذار نیست، رسیدند.

این مطالعات چندین مسئله درخصوص روش پژوهش داشتند. برای مثال، طول زمانی را که برای یادگیری برنامه‌نویسی صرف شده و نوع برنامه‌هایی که ساخته شده‌اند را درنظر نگرفته‌اند. این ویژگی‌ها در حال حاضر به‌عنوان خصوصیات ابزاری در طراحی آموزش‌های موفق برنامه‌نویسی [پالومبو[۱۹]، ۱۹۹۰] شناخته شده‌اند که به این موضوع بعداً در بخشی که به طراحی نرم‌افزار برای یادگیری اختصاص داده شده است، با جزئیات خواهم پرداخت. مسئلهٔ دیگر این بوده است که معلمان غالباً خود را با لوگو تطبیق داده‌اند، اما به نوآوری‌های پداگوژیکی برای یادگیری ریاضی و علوم دست نزده و حتی اگر نوآوری داشته‌اند، حمایت‌های آموزشگاهی گسترده‌ای در مدارسشان برای انجام چنان کارهایی دریافت نکرده‌اند [پَپرت، ۱۹۹۱ و ۱۹۹۷]. بیشتر نیروهای آموزشگاهی برای شکل دادن استفاده از لوگو در مدرسه به کار گرفته شده‌اند و نتیجه غالباً این بوده که ایده‌های ساختن‌گراها در مورد یادگیری و تدریس آخرین چیزی بوده که مورد توجه قرار گرفته است.

ایده‌های کلیدی در ساختن‌گرایی

نام ساختن‌گرایی استعاره‌ای از یادگیری را در ذهن متبادر می‌کند که در آن هرکس دانش خودش را می‌سازد و غالباً با آموزش‌های سنتی و دستورالعملی مقایسه می‌شود که استعاره‌ای از یادگیری با انتقال دانش را یادآوری می‌کند. اگرچه این دو استعاره مقایسه‌ای خلاصه‌وار از دو نوع یادگیری ارائه می‌دهند، اما این مقایسه وقتی ارزشمند می‌شود که ایده‌های ساختن‌گراها از ساختن دانش کاملاً تشریح شود و ویژگی‌های آن، چه در مورد افراد و چه در مورد گروه‌های پویا به دقت بررسی شوند. پس ما به سمت تفکرات این فرهنگ یادگیری حرکت و مشخص می‌کنیم که کدام ویژگی‌ها از این محیط یادگیری مشوق ساختن موفق دانش است.

ساختن دانش

این ایده که هرکس دانش خودش را می‌سازد، قطعاً از تئوری توسعهٔ دانش **پیاژه** و بینش ابزاری او که بچه‌ها دنیا را اساساً از راه‌های متفاوتی نسبت به بزرگسالان می‌فهمند، استخراج شده است. او دو سازوکار تشخیص داد: جذب[۲۰] و انطباق[۲۱] که چگونگی حس بچه‌ها از دنیا را وقتی که با آن تعامل می‌کنند و تجربیات خود را با فهمشان در می‌آمیزند، توضیح می‌دهد. ساختن‌گرایی براساس این دو سازوکار ساخته می‌شود و بر فرایندی که به یادگیرنده‌ها کمک می‌کند تا بین آنچه قبلاً یاد گرفته‌اند ارتباط برقرار کنند، تمرکز و تأکید می‌کند. یک وجه کلیدی در ساختن دانش این است که چگونه یادگیرنده‌ها دانش خودشان را می‌سازند و با آن شروع به تشخیص و درک و فهم می‌کنند. این ویژگی اختصاصی از هوشمندی نشئت می‌گیرد و شامل ارزش‌های اساسی است.

با توجه به نظرات پَپرت، اشیای فیزیکی در فرایند ساختن دانش نقش‌محوری و مرکزی دارند. او عبارت «اشیایی برای تفکر با» را به‌عنوان روشن‌کننده این موضوع ثبت کرد که چگونه اشیا در دنیای فیزیکی و دیجیتالی (مثل برنامه‌ها، روبات‌ها و بازی‌ها) می‌توانند اشیائی ذهنی شوند که کمک می‌کنند مفاهیم ساخته شوند، آزمایش گردند و ارتباط‌های بین دانش‌های قدیم و جدید را بازسازی و درباره آن تجدید نظر کنند. «اشیایی برای تفکر با»[۲۲]، از قبیل لاک‌پشت لوگو به‌طور خاص در حمایت از این

ویژگی اختصاصی تأثیرگذار هستند، چون آن‌ها توانایی بازشناخت بچه‌ها و یادگیری سازگار با محیط را آسان می‌کنند.

ساختن‌گرایی بیشتر از مدل پیاژه در خصوص برابری ارزش ملموس‌ها و انتزاع بین آن‌ها تفاوت قائل می‌شود. در مراحل نظریه [رشد شناختی] پیاژه، انتزاع رسمی به‌عنوان هدف غایی ساختن دانش دیده می‌شود، تفکر ملموس همیشه برای کوچک‌ترها در مقایسه با بچه‌های بزرگ‌تر در نظر گرفته می‌شود. **ترکل**[۲۳] و پَپرت (۱۹۹۰) در عوض مدعی شدند که تفکر ملموس می‌تواند حتی برتر از تفکر انتزاعی لحاظ شود. علوم به طور عمومی و فرهنگ رایانه به‌طور خاص، قصد داشته‌اند که تفکر انتزاعی را ارزشمند جلوه دهند. اما در مطالعهٔ برنامه‌نویس‌ها، ترکل و پَپرت کشف کردند که ترغیب کردن رسمی و از بالا به پایین رویکردهای برنامه‌ریزی همیشه به بداهه‌پردازی‌های عالی منجر نمی‌شود و بیشتر به نگرش‌های غافلگیرانه و ناگهانی شبیه است. در سبک‌های غافلگیرانه مراحل مشخص و غیرقابل تغییر به سمت صورت‌های عالی از ساختن دانش نیستند اما به راه‌هایی کیفی متفاوت از روش‌های سازمان‌دهی، طرح‌ریزی یا حل مسئلهٔ فردی شبیه هستند.

به‌طور خلاصه، ساختن دانش عبارت است از: «کنکاش کردن در بخشی از یادگیری که شامل ساختن ارتباط‌هایی بین موجودات ذهنی که از قبل موجود بودند، با موجودات ذهنی جدیدی است که به نظر می‌رسد با دقت بیشتری به وجود آمده‌اند و از کنترل پیوستهٔ آگاهی فرار کرده‌اند ... این موضوع یک راهبرد پیشنهاد می‌کند: یادگیری با ارتقا در محیط‌های یادگیری و همچنین به وسیلهٔ عمل کردن فرهنگ‌ها (بیشتر از افراد) تسهیل می‌شود» (پَپرت، ۱۹۹۳: ۱۰۵)

در ادامه نویسنده به تشریح ویژگی دیگر نظریه ساختن‌گرایی با عنوان «فرهنگ یادگیری» می‌پردازد. با ذکر مثال‌های متفاوت از جمع‌های بزرگسالان و دانش‌آموزان که هر یک پروژه‌ها و دغدغه‌های خود را در یادگیری دنبال می‌کنند و در واقع برنامه درسی مشخصی وجود ندارد، به این نکته اشاره دارد که این نوع نگاه به یادگیری به همراه خود فرهنگی را تولید می‌کند که متفاوت از فرهنگ معمول و متداول مدرسه و کلاس‌های سنتی است. این مثال‌ها ما را به تدریج به سمت استفاده از رایانه در این نوع فرهنگ یادگیری پیش می‌برد تا زمینه برای طرح موضوع دیگری که به‌طور گسترده به این تئوری مربوط می‌شود، فراهم شود.

در واقع برای محقق شدن ایدهٔ ساختن‌گرایی به محیطی (مجازی یا فیزیکی) نیاز است که امکان ساختن را برای یادگیرنده فراهم کند. در اینجا دو مفهوم مطرح می‌شود:

۱. محیط‌های شبیه‌سازی دیجیتالی که در ادبیات پژوهشی مرتبط با ساختن‌گرایی از آن با عنوان «دنیاهای کوچک»[۲۴] یاد می‌شود.

ویژگی مهم برنامه‌نویسی لوگو این ایده بود که بچه‌ها دربارهٔ چگونه فکر کردن و یادگیری خود، یاد می‌گرفتند، که بازتاب یا «فراشناخت» نامیده می‌شد

۲. بسته‌های دست‌ساز که شامل قطعات فیزیکی‌اند و شرایط را برای دست ورزی یادگیرنده فراهم می‌کنند. (کیت‌های ساختن[۲۵]).

اصطلاح دنیای کوچک به یک محیط تعاملی رایانه‌محور برای یادگیری اشاره دارد که پیش‌نیازهای موردنظر برای ساختن در آن محیط رایانه فراهم است و جایی است که یادگیرنده فعال می‌شود و چگونگی یادگیری خود را می‌سازد. همچنین یکی از بسته‌های دست‌ساز معروف و شناخته‌شده «لگو»[۲۶] نام دارد که شامل قطعات مستطیل‌شکل (آجرها) موتورها، حسگرها و پردازشگر است. با نوشتن برنامه روی پردازشگر، این امکان فراهم می‌شود که ترکیبی از ساختن فیزیکی و دیجیتالی فراهم شود.

با توجه به موارد فوق، یکی از مهم‌ترین مباحث پژوهشی مرتبط با ساختن‌گرایی، طراحی فعالیت‌ها و تأثیر آن بر یادگیری است. در پایان این مطلب یادآور می‌شوم که با محوریت نظریه ساختن‌گرایی و محیط‌های شبیه‌سازی مثل لوگو و سایر محیط‌های رایانه‌ای، هزاران پژوهش انجام شده‌اند و هنوز ادامه دارند. شایان ذکر است که این موضوع و شاخهٔ کلی‌تر آن (استفاده از ابزارهای دیجیتالی در آموزش ریاضی) هنوز جای خود را در فضای دانشگاهی و تخصصی رشته آموزش ریاضی باز نکرده‌اند، در حالی که تعداد مقالات و پژوهش‌ها در این زمینه در خارج از ایران روز به روز رو به افزایش است. امید دارم این مطلب فتح بابی باشد برای تحقیقات بیشتر در این حوزه آموزشی و شاهد کاربست‌های آن در مدرسه‌ها و آموزش‌های رسمی باشیم.

پی‌نوشت‌ها

1.Yasmin B. Kafai
2. The Cambridge Handbook of The Learning Sciences
3. R. Keith Sawyer
4. Seymour Papert
5. Logo
6. Jean Piaget
7. Marvin Minsky
8. Mind storm
9. Instruction
10. evocative object
11. Sherry Turkle
12. Basic
13. PEN DOWN
14. Syntonic
15. Richard Noss
16.Celia Hoyles

17. Roy Pea
18. Midian Kurland
19. Palumbo
20. assimilation
21. accommodation
22. object-to-think-with
23. Turkle
24. microworlds
25. construction kits
26. Lego

منابع

1. Ching, C. C. (2000). *Apprenticeship, Learning and technology: Children as oldtimers and newcomers in the culture of learning through design*. Unpublished doctoral dissertation. University of California, LosAngeles.
2. Disessa, A. (2000). *Changing minds: Computers, Learning and Literacy.* Cambridge, MA: MIT Press.
3. Edwards, L. (1998). Embodying mathematics and science: Microworlds as representations. *Journal of Mathematical Behavior*. 17(1).
4. Eisenberg, M. (2003). Mindstuff: Educational technology beyond the computer: *Convergence*, 4.
5. Furth, H. G. (1987). *Knowledge as desire: An essay on Freud and Piaget*. NewYork: Columbia University Press.
6. Harel. I. (1990). *Children designers.* Norwood, NJ: Ablex.
7. Harel, I., & Papert, S. (1991). Software design as a learning environment. *Interactive Learning Environments*, 1(1).
8. Kafai, Y. B. (1995). *Minds in play: Computer game design as a context for children's learning.* Hillsdale, NJ: Lawrence Erlbaum Associaties.
9. Kafai, Y. B. (1996). Gender differences in children's constructions of video games. In Patricia M. Greenfield & Rodney R. Cocking (Eds.), *Interacting with video* (pp. 39-66). Norwood, NJ: Ablex publishing Corporation.
10. Kafai, Y. B., & Ching, C. C. (2001). Affordances of collaborative software design planning for elementary students' science talk. *The Journal of the Learning Sciences*, 10(3).
11. Kafai, Y. B., Franke, M., Ching, C., & Shih, J. (1998). Games as interactive learning environments fostering teachers' and students' mathematical thinking. *International Journal of Computers for Mathematical Learning*, 3(2).
12. Kafai, Y. B., Resnick, M. (1996). *Constructionism in practice*. Mahwah, NJ: Lawrence Erlbaum Associates.
13. Kafai, Y. B., & Roberts, M. (2002). On becoming junior software designers. In R. Stevens & P. Bell (Eds.), *Proceedings of the Fifth International Conference on the Learning Sciences* (pp.191-198). Mahwah. NJ: Erlbaum.
14. Keller, E. F. (1983). *A feeling for the organism: The life and work of Barbara McClintock.* San Francisco: W.H. Freeman.
15. Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation.* London: Cambridge University Press.
16. Marshall, S. (2000). *Planning in context: A situated view of children's management of science projects.* Unpublished doctoral dissertation. University of California, Los Angeles.
17. Martin, F. (1996). Ideal and real systems: A study of notions of control in undergraduates who design robots. In Y. Kafai & M. Resnick (Eds.), *Constructionism in practice* (pp. 255-268). Mahwah, NJ: Lawrence Erlbaum Associates.
18. Martin, F. (2001). *Robotic Explorations: A Handson introduction to engineering*. NewYork: Prentice Hall.
19. Noss, R., Hoyles, C. (1996). *Windows on mathematical meanings: Learning cultures and computers.* Dordrecht: Kuwer Academic Publishers.
20. Palumbo, D. (1990). Programming language/ problem-solving research: A review of relevant Issues. *Review of Educational Research*, 45.
21. Papert, S. (1980/1993). *Mindstorms* (2nd ed.). NewYork: Basic Books.
22. Papert, S. (1987). Computer criticism versus technocentric thinking. *Educational Researcher*, 16(1).
23. Papert, S. (1991). Situating constructionism. In I. Harel & S. Papert (Eds.), *Constructionism* (pp. 1-14). Hillsdale, NJ: Lawrence Erlbaum Associates.
24. Papert, S. (1993). *The children's Machine: Rethinking school in the age of the computer*. NewYork: Basic Books.
25. Papert, S. (1997). Tinkering towards utopia: A century of public school reform. *Journal of the Learning Sciences,* 6(4).
26. Pea, R., & Kurland, M. (1984). On the cognitive effects of learning computer programming. *New Ideas in Psychology,* 2(2).
27. Resnick, M. (1991). New paradigms for computing, new paradigms for thinking. In Y. Kafai & M. Resnick (Eds.), *Constructionism in Practice* (pp. 255-268). Mahwah, NJ: Lawrence Erlbaum Associates.
28. Resnick, M. (1994). *Turtles, Termites, and Traffic Jams*. Cambridge, MA: MIT Press.
29. Resnick, M. (1998). Technologies for life long learning. *In Educational Technology, Research Development,* 46(4).
30. Resnick, M., & Ocko, S. (1991). LEGO/Logo: Learning through and about design. In I. Harel & S. Papert (Eds.), *Consructionism* (pp.141-150). Hillsdale, NJ: Lawrence Erlbaum Associates.
31. Resnick, M., Rusk, N., & Cooke, S. (1998). The computer clubhouse: Technological fluency in the inner-city. In D. Schon, B. Sanyal, & W. Mitchell (Eds.), *High Technology and Low Income Communities* (pp. 266-286). Cambridge, MA: MIT Press.
32. Resnick, M., & Wilensky, U. (1998). Diving into complexity: Developing probabilistic decentralized thinking through role-playing activities. *Journal of the Learning Sciences,* 7(2).
33. Turkle, S. (1995). *Life on the screen: Identity in the age of the Internet.* NewYork: Simon & Schuster.
34. Turkle, S., & Papert, S. (1990). Epistemological pluralismand the reevaluation of the concrete. *Signs*, 16(1).