Ingenium: Engaging Novice Students with Latin Grammar

Sharon Zhou[1]   **Ivy J. Livingston**[2]   **Mark Schiefsky**[2]   **Stuart M. Shieber**[3]   **Krzysztof Z. Gajos**[3]

[1]Harvard College
Cambridge, MA USA
sharonz@post.harvard.edu

[2]Harvard Dept. of Classics
Cambridge, MA USA
{livings,mjschief}@fas.harvard.edu

[3]Harvard SEAS
Cambridge, MA USA
{shieber,kgajos}@seas.harvard.edu

ABSTRACT

Reading Latin poses many difficulties for English speakers, because they are accustomed to relying on word order to determine the roles of words in a sentence. In Latin, the grammatical form of a word, and not its position, is responsible for determining the word's function in a sentence. It has proven challenging to develop pedagogical techniques that successfully draw students' attention to the grammar of Latin and that students find engaging enough to use. Building on some of the most promising prior work in Latin instruction—the Michigan Latin approach—and on the insights underlying block-based programming languages used to teach children the basics of computer science, we developed Ingenium. Ingenium uses abstract puzzle blocks to communicate grammatical concepts. Engaging students in grammatical reflection, Ingenium succeeds when students are able to effectively decipher the meaning of Latin sentences. We adapted Ingenium to be used for two standard classroom activities: sentence translations and fill-in-the-blank exercises. We evaluated Ingenium with 67 novice Latin students in universities across the United States. When using Ingenium, participants opted to perform more optional exercises, completed translation exercises with significantly fewer errors related to word order and errors overall, as well as reported higher levels of engagement and attention to grammar than when using a traditional text-based interface.

Author Keywords

Engagement; Grammar; Latin education.

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

INTRODUCTION

Reading Latin poses many difficulties for English speakers, because they are accustomed to relying on word order to determine the functional roles of words in a sentence [40, 54]. In Latin, it is the grammatical form of a word that is responsible for determining its function in a sentence, and not its

position. For example, *agricolae invenerunt pueros* translates as "the farmers found the boys", while *agricolas invenerunt pueri* is "the boys found the farmers". The two sentences are composed of the same words in the same order, but the different grammatical forms of the words *agricola* (farmer) and *puer* (boy) are responsible for the different meanings.

Despite the necessity of distinguishing between such morphological forms in reading, conventional Latin textbooks [61, 62, 3, 28] tend to accelerate their study, quickly introducing complex sentences with the long-term goal of transitioning students to reading original texts. However, this approach comes at the cost of instilling anxiety in students around grammar and fails to provide students with an adequate grasp of key grammatical concepts to apply to reading [29, 41].

Typically, these textbooks develop knowledge of morphology through repetitive drill exercises. These drills, however, practice words only in isolation [29], which is insufficient preparation for reading complex texts, because word forms are often ambiguous and students must use context to guide their interpretations. Thus, interactions between words in a sentence are extremely important, and pedagogical scholarship has urged development of methods to teach students how to identify the interactions in order to read Latin effectively [52, 20, 12].

In the 1950's, researchers at the University of Michigan designed a relevant pedagogical technique, called the Michigan Latin approach, that addressed the challenge of students' over-reliance on word order and their inadequate attention to grammatical forms and interactions. While studies demonstrated that the Michigan Latin approach was successful in increasing reading comprehension [40], it has not been widely adopted, possibly because it relies on idiosyncratic technical jargon and does not engage students with the material any more than traditional methods [49].

How might we leverage the strengths of the Michigan Latin approach, while minimizing its shortcomings? We found inspiration in block-based programming languages, particularly Scratch [37, 19, 44, 39, 23, 43, 33]. With an interactive and intuitive interface, Scratch is able to effectively engage novices [37]—a feat that the Michigan Latin approach was not able to achieve.
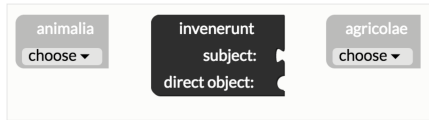
(a) The student encounters the sentence *animalia invenerunt agricolae*, meaning "the farmers found the animals". The sentence is accompanied by Ingenium blocks, horizontally arranged in the same order as the words in the sentence.
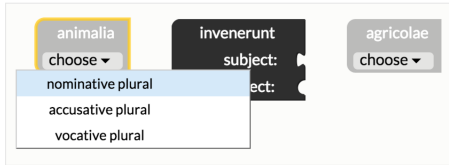
(b) The student inspects possible forms of *animalia* and, guided by word order, assumes that the word should be interpreted as being in the nominative case, which corresponds to the role of the subject.

(c) Assigning a grammatical role to *animalia* causes it to change color and to acquire a knob shape corresponding to its case. Because of the grammatical match, the student is able to attach *animalia* to the subject slot of the verb.
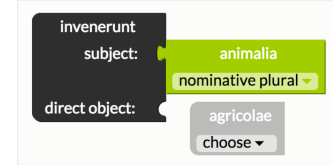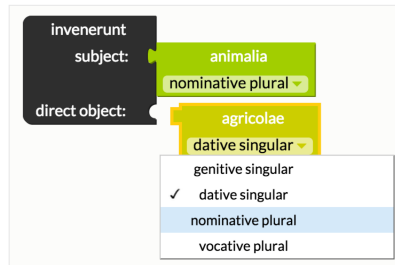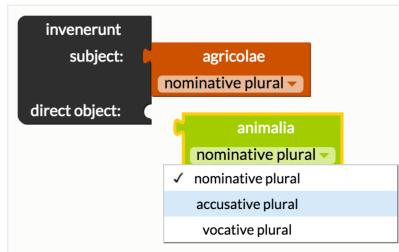
(d) At this point, the verb still needs a direct object. The student realizes, however, that *agricolae* can not be in the accusative case, which corresponds to the role of the direct object.

(e) The student backtracks: she removes *animalia* from the subject role and instead assigns nominative case to *agricolae* and places it into the role of the subject. She then assigns the accusative case to *animalia*.
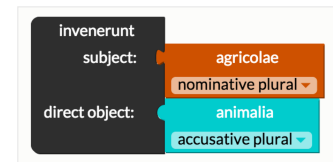
(f) She places *animalia* into the role of the direct object of the verb. The completed puzzle reveals a grammatically valid configuration of the sentence. It is now her responsibility to derive meaning from her understanding of the grammar.

Figure 1: Sample interaction with Ingenium.

Building on the strengths of the Michigan Latin approach and Scratch, we developed Ingenium[1]. Like Scratch, Ingenium uses a visual language, but instead of building new sentences, Ingenium enables learners to visually reconstruct relationships among components of an existing sentence as illustrated in Figure 1. The design of Ingenium addresses the goal of the Michigan Latin approach to focus students' attention on the interactions between words. Ingenium visualizes Latin words as puzzle pieces, whose shapes and colors indicate each word's grammatical functions. As implied by their matching shapes and colors, blocks fit together if their grammatical relationship is sound. However, because the same spelling of a word can often correspond to several possible grammatical forms, students have meaningful choices to make regarding the roles that different words play in a sentence.

Unlike distinct programming concepts whose differences remain consistent in various contexts, forms of Latin words that differ in function can look identical. The correct choice of form depends on the grammatical structure underlying the rest of the sentence, or the context of all words in the sentence. To reflect this, we designed blocks to be mutable so that students could play with each of their forms, and thus each of their roles, in the sentence, so that they may satisfy the constraints of each word in relation to the constraints of all the others.

Importantly, Ingenium is designed so that completing a puzzle does not give away a translation—it only reveals a grammatically permitted configuration of roles for the words in the sentence. Thus, we expect students to reflect more on the grammar, as revealed by the morphology of the words, instead of on the word order.[2] While the constraints of the system allow students to find and verify each word's grammatical role, students are still left with the responsibility of turning these roles into meaning.

We adapted Ingenium to be used for two standard classroom activities: sentence translations and fill-in-the-blank ex-

---

[1]On the naming of Ingenium, the word *ingenium* in Latin means "innate quality", "invention", and "genius", from which we derive the English words "ingenuity" and "engineer".

[2]We use the term "reflection" here and throughout in its non-technical meaning of "serious thought or consideration". We do so consistently with the attention to grammar as considered by the Michigan Latin approach and as it would be understood by the student participants in the study, who were asked about their reflection, as opposed to in its technical sense that is used by educational psychology theorists.

ercises. We evaluated Ingenium with 67 novice Latin students in universities across the United States. In a within-subjects study, participants performed translation and fill-in-the-blank exercises with and without Ingenium. Students were deemed successful when they were able to correctly interpret the meaning of Latin sentences. Hence, they could not "game the system" by simply completing the puzzles through trial and error, because solving the blocks-based puzzle was only a means toward uncovering the semantics of the sentence. When using Ingenium, participants opted to perform more optional exercises, completed Latin translation exercises with substantially higher accuracy—particularly with fewer errors related to word order—and reported higher levels of engagement and attention to grammar than when using a traditional text-based interface.

RELATED WORK

The Michigan Latin Approach

The University of Michigan created a new set of instructional materials in the 1950's that focused students attention on the grammatical interactions among words necessary to reading a Latin sentence [49, 54, 55, 31, 50]. They trained students to master conceptual techniques instead of repetitive drills, an effective strategy in improving Latin grammar comprehension [35, 27, 18].

The Michigan Latin approach calls its core technique "metaphrasing" [40]. Students apply metaphrasing to reading Latin sentences in order to narrow down the number of possible roles that each word in a sentence can assume. The technique has students analyze and deconstruct each word in a sentence into its possible functions and to consider how each of these functions might predict or affect the functions of other words in the sentence [49, 40]. Students then lay out in English word order the possible configurations of a sentence given their analyses [49]. Through metaphrasing, students are trained to anticipate a constrained set of feasible roles that the remaining words can take. These anticipated predictions, like the blanks in a fill-in-the-blank exercise, form "gaps" in a sentence, and guide students to a grammatically sound interpretation [49, 40].

The approach addresses the common mistake among beginning and more advanced students alike of focusing too little attention on the grammatical construction of a sentence and relying too heavily on word order and isolated dictionary definitions. It teaches students to think consciously about a Latin sentence's grammar before translating, and to use the grammar of the language, not their preconceptions such as word-order dependence, to guide their interpretations [49].

The effectiveness of the Michigan Latin approach lies in breaking down the grammatical construction of a sentence into more manageable parts and establishing a structured and dependable approach to understanding each part [40]. However, the approach fails to capture Latin's flexible ordering of words and thus may not fully remove the influence of word order, a limitation that the creator of the Michigan Latin approach has acknowledged [49]. In fact, second language instructional theory has shown the importance of abstract restructuring and explored how more advanced students, as opposed to novices, newly introduced to a language, restructure parts of a sentence into abstract schemata [42].

Scratch and Other Block-Based Languages

Removing the complexities of a language's syntax and vocabulary, Scratch uses puzzle blocks of varying shapes and colors to represent the concepts, with brief descriptors to help students quickly identify concepts [38]. Instead of memorizing a concept or how it works, students can focus their attention on the concept's application and interaction with other concepts. By using an interactive visual approach both that focuses on the structure behind the concepts and that removes the intimidating concrete syntax of a textual language such as Java [37, 38, 19], Scratch excites, engages, and effectively introduces novices to elementary programming [44, 39, 23, 43, 33, 63, 37].

Another key feature of Scratch is that students cannot make syntactic mistakes in the system, because blocks only fit together if their concepts match [37, 38]. Without the discouraging error-prone environment of many text-based programming languages, Scratch encourages students to experiment with the meaning, while constraining their expression to syntactically-correct programs. Ultimately, Scratch's intuitive environment, the embodiment of the "recognition rather than recall" principle of direct manipulation [26], focus on broad concepts, and prevention of student-made errors make it succeed in engaging novice programmers.

In addition to Scratch, many other block-based languages have been developed for students to learn basic programming, such as Blockly [16], Alice [14], MIT App Inventor [47], Snap! [22], RoboBuilder [60], Storytelling Alice [30], Pencil Code [6], and Droplet [5].

Additional applications of block-based languages have emphasized modeling concepts and substituting code: Polymorphic Blocks that visualize the logic of mathematical proofs and formulas [34], BlockImpress that creates online slideshow presentations with static images, text, charts, graphs, and formulas [56], Pixly that alters images through pixel-level operations [58], Spherly that programs a Sphero robot [58], FabCode that programs 3D models for fabrication such as printing and laser cutting [1], StarLogo TNG that facilitates 3D game design [59], a language that builds SPARQL queries for querying Linked Data [7], and several languages that enable software game design without code, such as Stencyl [36], LEGO Mindstorms [2], Etoys [53], Hopscotch [25], and Pocket Code [10].

We introduce Ingenium as a novel application of a block-based language to natural-language learning and acquisition. Ingenium is also distinctive in that it uses a block-based language to reverse engineer the meaning of an existing artifact, rather than to support learners in constructing new ones.

Ingenium was adapted from the block-based language Blockly [16].

Interactive Language-Learning Technologies

Recent approaches to interactive language learning have focused on relevant vocabulary placement for learning new ter-

minology, such as displaying relevant keywords alongside articles [21], crowdsourcing colloquialisms to native speakers [11], presenting interactive subtitles for learning vocabulary in foreign films [32], and prompting users to practice timely, relevant vocabulary quizzes as they wait for a response from an online chat partner [8]. Gamified language-learning environments, such as Duolingo or Quia word games, have been studied to improve students' motivation [57, 13, 15]. Technology aimed to teach Mandarin Chinese has abstracted out text, using images as definitions [21], or associating spoken words with their pictographic representations [17].

Ingenium, by comparison, moves away from vocabulary recall to emphasize the importance of grammatical structure in informing students of a sentence's meaning. Ingenium associates the text of a word with the abstract visualization of its possible grammatical roles. The interactive, hands-on, and dynamic interface of the blocks, in addition to the positive feedback of snapping together matching blocks, encourages the user to engage with it and gives it a game-like nature.

INGENIUM

We have designed Ingenium with two core objectives in mind: (1) to help students understand the meaning of Latin sentences by focusing their attention on their grammatical structure, and (2) to create an engaging user experience for exploring the complex structure of Latin grammar.

Ingenium allows users to visually contrast possible roles of a Latin word and to explore interactions among different words in a hands-on manner with puzzle blocks representing Latin words. The jigsaw metaphor helps students reason about constraints on each word's role in a sentence, while the mutability of the pieces allows them to realize the range of roles that each word can embody given its spelling. The system prevents syntactically incorrect parsings, giving students the license to experiment without fear of making grammatical mistakes. Additionally, by removing specialized terminology in favor of visual affordances, Ingenium is designed to engage students with Latin grammar.

The act of involving all the blocks focuses students attention on the process of assigning correct grammatical roles to words in a complex Latin sentence. The assignment of grammatical roles aids students in uncovering the correct meaning of the sentence. The success of Ingenium rests in its ability to help students reach a semantic understanding of a sentence by focusing their attention more on the grammar.

Formative Design Research

We designed Ingenium through an iterative process. We started by creating paper prototypes of puzzle blocks from colorful 5 x 8-inch index cards. With them, we experimented with the visualization of the following grammatical concepts:

- Different valid cases of nouns with the same morphology
- Nominative subject and verb agreement
- Accusative case as the direct object
- Adjective and noun agreement
- Prepositional phrases
- Verb transitivity

- Use of linking verbs

- Subordination of relative clauses

We explored two main designs. In one design, shape primarily informed function and color indicated the part of speech. In the other design, the reverse was true: color informed function and shape indicated the part of the speech. In the first design, we explored two sub-designs on the subordination or juxtaposition of an adjective in its relationship with a noun: adjectives could become embedded into a noun or they could attach themselves adjacent to an agreeing noun. All designs enforced a set subject-verb-object word order. Not unlike "gapping" in the Michigan Latin approach, they constructed sentences in a horizontal and linear structure.

We asked instructors at a local Montessori School, who were familiar with pedagogies that use color-coded blocks to represent parts of speech in English, and students at our university with high-school level Latin to play freely with 30 puzzle blocks and the two separate designs.

We observed that shapes more effectively communicated grammatical connections between blocks than colors when there were fewer shapes and clear differentiation among them. As the number of shapes and colors grew, colors performed as more effective indicators. This informed our designs to employ shapes for important indicators that had fewer combinations, such as part of speech and cases, while using color to indicate the many case-number-gender combinations.

Subordinating adjectives was less intuitive initially, yet was preferred once discovered. It allowed participants to group blocks together, and thus to organize them and understand that there were clear separation of roles. We thus observed that different kinds of interactions between parts of speech could effectively convey differences in their relationships.

We found that the grammatical roles of individual words caused more problems than the structural construction of clauses, causing us to focus our subsequent designs on the roles of each word. Since a word's role in a Latin sentence is chiefly determined by its part of speech, we decided to use part of speech to guide the main shape of blocks in order to emphasize grammatically valid connections among words in the design of our system. Ingenium currently illustrates interactions among the following five most common parts of speech, whose interactions are taught earliest in the classroom: nouns, verbs, adjectives, prepositions, and adverbs. (See full interactions in Figure 2.)

Usability Tests

We performed early usability testing of the initial software implementation of the system with five volunteers whose knowledge of Latin ranged from novice to expert.

All parts of speech in both our formative design research and the initial implementation were depicted with colorful blocks. We discovered that applying color to both types of blocks, those that change in color and those that remain the same in color, generated confusion. Communicating the neutrality of the immutable blocks (verbs, prepositions, adverbs) through

(a) Students are provided these blocks with the Latin sentence that means: "After the deed, fortune willingly gives plenty to the weary soldiers." The verb *dat*, "gives," and preposition *post*, "after," are designed with inline knob cutouts for the nouns that they expect to take. Note that the noun *copiam*, "plenty," is not an ambiguous form, so it has no dropdown menu of alternatives, unlike the other nouns and adjectives that have "choose" written on them.
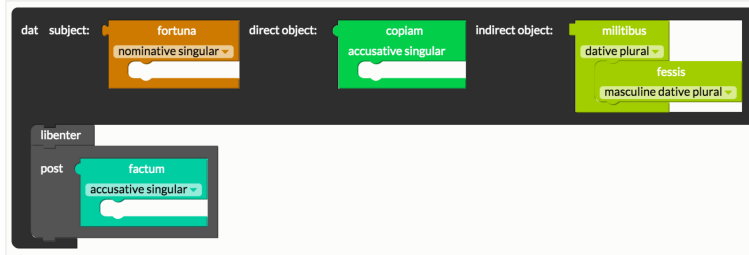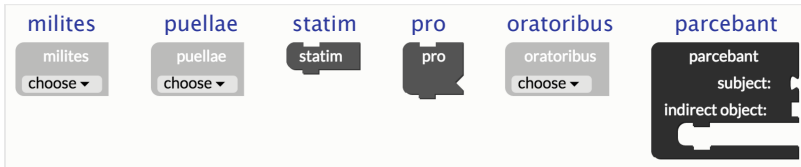

(b) The blocks for the inline design are completely matched. Inside the clamp of the verb *dat*, the adverb *libenter*, "willingly," and the prepositional phrase *post factum*, "after the deed," are nested and stacked. Similarly, inside the clamp of the noun *militibus*, "soldiers," the adjective *fessis*, "weary," is nested.

(c) This sentence, meaning "The soldiers immediately spared the girl on behalf of the orators," illustrates the external design of the verb *parcebant* and the preposition *pro*.
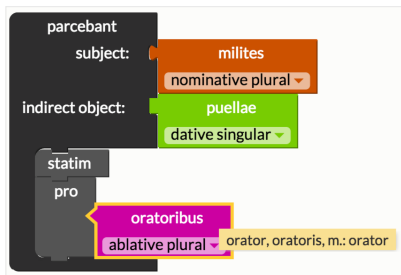
(d) The completed sentence with the external design shows how the verb block expands vertically to accommodate its expected noun blocks. The "clampless" nouns here are to prevent confusing students with unused visual structures, before they have been exposed to the appropriate interactions. Students can also gloss words at any point by right-clicking on a block, such as *oratoribus*, which then shows a text box with the word's dictionary entry, "orator, oratoris, m.: orator," where it was clicked.

Figure 2: Comparison between inline and external designs, and illustration of interactions among all parts of speech.

gray hues (black, charcoal gray, and light gray) reduced this confusion.

One user's embarrassment over forgetting the definition of a word prompted him to stop using the system, while other users were eager to test more blocks. This observation motivated our decision to include dictionary glossing by right-clicking a block, to shift the student's attention to the grammatical, and away from the vocabulary, questions.

Nouns and Verb-Noun Interactions
In Latin, the function of a noun in a sentence is often ambiguous, because a single spelling can correspond to several declined forms—information that a noun carries on its case, number, and gender. Nouns of each case can indicate several different functions, so a single form of a noun can also assume multiple roles. For example, in Figure 2a, *copiam* in the role of direct object could also, as its knob shape suggests, fit inside the preposition *post* to mean "after plenty." Students are responsible for deciding the correct role of the noun by using

contextual information about the grammatical constraints of both the noun itself and other words in the sentence.

Verb-noun interactions form the foundation of simple complete sentences. In Ingenium, we make this interaction visually apparent by having verbs and nouns connect like puzzle pieces through matching knob shapes. The noun block has a horizontally projecting knob on its top left corner that, if grammatically appropriate, can fit inside the same shape cut into a verb. (See the nouns *agricolae* and *animalia* fitting the verb *invenerunt* in Figure 1f.) Knob shapes correspond to one of the six declined cases: nominative, genitive, dative, accusative, ablative, or vocative. Through a dropdown menu, students are able to select from all the case-number combinations of the noun that are consistent with its spelling. (See Figures 1b, 1d, and 1e.) Before a selection is made, the block does not have a knob, is a neutral gray color, and has the word "choose" over the dropdown menu, so as not to bias the student's decision. (See *animalia* and *agricolae* in Figure 1a.) Once a combination is chosen and the case-number

combination is established, the corresponding knob appears dynamically in the top left hand corner. Students can change the case-number combination of a noun at any time to dynamically change the noun's knob shape and thus its function. (See *animalia* from Figure 1d to 1f.)

Noun-Adjective Interactions
Because a noun's case is the primary factor in determining its possible functions, we wanted to make interactions associated with the noun's case most apparent to the student. The noun's case thus determines the shape of the knob. Along with its case, however, a noun's gender (masculine, feminine, or neuter) and number (singular or plural) indicate which adjectives can modify it, as nouns can only be modified by adjectives of the same case-number-gender combination. A unique color is assigned to each of 36 possible case-number-gender combinations such that agreeing nouns and adjectives have the same color. (See *militibus* and *fessis* in Figure 2b.)

Structurally, the noun-adjective interaction is facilitated by the tooth-shaped concavities above and convexities below on adjective blocks and inside clamp structures beneath the body of noun blocks. (See *fessis* in Figure 2b.) Agreeing adjectives can be stacked unboundedly on top of each other—a grammatically permissible, though semantically improbable, connection—as the enclosing clamp expands to accommodate them.

Verbs, Prepositions & Adverbs: Two Designs
The expectations raised by the verb in a sentence are often the most constraining and thus provide the most helpful information in determining the roles of other words and the overall grammatical structure of the sentence. The knob cutouts in a verb block indicate, implicitly through its shape, the case and, explicitly with text, the roles that the given verb expects. For example, the intransitive verb *parcebant* in Figure 2c has special properties in Latin and expects an object in the dative case, as well as a subject in the usual nominative case. The cutouts in the verb blocks were motivated by the "gapping" concept in the Michigan Latin approach as students form appropriate expectations for blanks, or "gaps," to fill when analyzing the verb's expectations.

We created two designs for the knob cutouts in verbs: *inline* and *external*. The two designs affect the knob cutouts, as shown in *dat* in Figure 2a and *parcebant* in Figure 2c. The inline design has the knob cutouts lie inside the main body of the block in a horizontal and linear layout, like blanks in a fill-in-the-blank exercise. The enclosure of the noun inside of the verb visually suggests that the form of the noun depends on the verb and, in turn, that the verb takes a particular form of the noun. However, the horizontal arrangement of the cutouts in the inline design may inadvertently convey a suggestion of expected word order. In order to minimize any suggestion of word order and to focus students' attention on the roles of words instead, we created the external design with the knob cutouts aligned vertically beneath the body of the block. This design sacrifices clear affordances of the cutouts, but also deemphasizes the word order. This design was prompted by the shortcomings of fixed horizontal word order in the Michigan Latin approach [49].

Similar to verbs, prepositions take nouns of a certain case as their objects and thus have knob cutouts identical to the ones on verbs. (See *post* in Figure 2a and *pro* in Figure 2c.) However, they only have one functional expectation for their dependent noun, so no textual indication of function is necessary. In order to maintain visual consistency with the inline or external design on verbs, students see the same design on prepositions.

The verb clamp is designed like the noun clamp to enclose arbitrarily many components such as prepositional phrases and adverbs that fit grammatically within the verb phrase. (See *dat* in Figure 2a and *parcebant* in Figure 2b.) To ease students into the use of clamps, the nouns and the verbs are initially "clampless". (See clampless nouns *milites*, *puellae*, and *oratoribus* in Figures 2c and 2d.) Only after the corresponding optional components have been shown and the interaction with them becomes necessary are the clamps necessarily introduced. Furthermore, we have designed the noun and verb clamps to have different teeth shapes, circular and rectangular, in order to visually distinguish the components that can be enclosed by the verb as opposed to the noun. Accordingly, prepositions and adverbs have circular and rectangular teeth, respectively, to correspond in shape with the clamps that enclose them.

Unlike the roles of declined nouns and adjectives, the expectations of a conjugated verb, preposition, and adverb are unambiguous. As changes in color reflect changes in declension, we were informed through usability tests to color the static verb and the preposition and adverb blocks respective shades of neutral black and charcoal gray in order to differentiate them from the mutable noun and adjective blocks. (See static coloring of the verb *dat* in black and the adverb *libenter* and the preposition *post* in charcoal gray in Figures 2a and 2b.)

Interacting With Blocks
When students interact with blocks, grammatical relationships are reinforced with visual highlighting, sound, and automatic snapping of matching pieces. If two blocks do not match, and if the student drops a block near another block with which it shares no connection, the dropped block will be bumped away from the other block to indicate that the connection is not grammatically valid and therefore cannot occur. Students can thus experiment freely without fear of making grammatical errors. Their interactions with Ingenium emphasize how a sentence is grammatically constructed, based on the forms and functions of words and independent of their order.

Ingenium also allows students to see a word's dictionary definition by right-clicking on its block, shifting students' attention to the grammatical problems and away from vocabulary questions. This decision was informed by formative studies in which users were not familiar with all of the words used in the system, making the system less welcoming and engaging.

Classroom Activities
We explored several applications of our system to existing classroom activities.

fill-in-the-blank

choose the correct form of *salus* to complete the sentence

nota bene: discard pieces added to the puzzle space by returning them back to the gray area
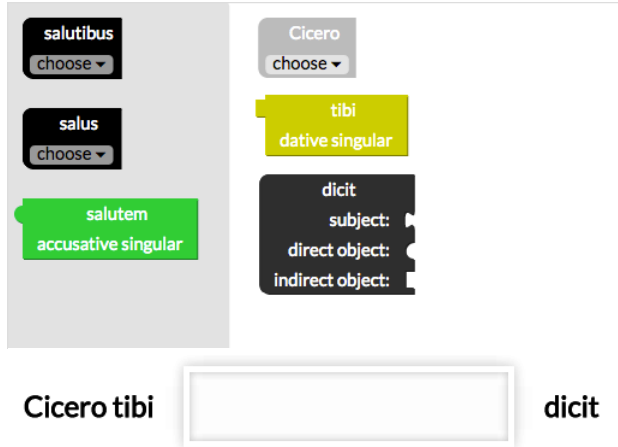

Figure 3: Sample fill-in-the-blank exercise with Ingenium: *Cicero tibi _____ dicit*. With the blocks in the gray zone on the left as options, students are free to experiment by dragging them on and off the white canvas to the right. Students can then type the grammatically valid option into the blank.

First, Ingenium applies particularly well to typical Latin-to-English translation exercises. Given a Latin sentence and the given blocks appearing in the same horizontally linear order of the words in the sentence, students can uncover a syntactically correct parsing of the sentence by fitting all the blocks together so that no word is left unattached, or without a grammatical role.

Another common exercise in Latin homework and exam questions is a fill-in-the-blank activity with multiple choice answers, one of which completes the sentence prompted in the question. Inspired by the concept of "gapping" in the Michigan Latin approach, Ingenium is also particularly suitable for this type of task (Figure 3). When given the exercise, the blocks that correspond to the words in the question are present on a main canvas on which students can freely play, and the blocks representing the multiple choice options for the blank appear in a word bank alongside the canvas. From this bank, students can drag the multiple choice blocks onto the canvas to experiment. They can also return the multiple choice blocks back to the word bank.

We also explored a sandbox interface, not deployed for the experiment, in which students can manipulate a wide array of blocks—deleting and adding new ones from a toolbox of various words—to construct grammatically correct Latin text that is meaningful to them.

EXPERIMENT

We conducted an experiment to evaluate Ingenium in comparison to traditional text-only presentation of Latin translation and fill-in-the-blank exercises. We focused in particular on the impact of Ingenium on novice Latin students' engagement, level of attention to grammar, self-efficacy, accuracy in translation, errors related to word order, and overall preference. We define novice to have 1/2 year to 11/2 years of prior college or high school Latin experience.

Hypotheses

The interactive, hands-on, and game-like nature of Ingenium makes it distinct from traditional language comprehension drills and we expected that this change would result in increased *engagement*. Engagement is important in enabling students to improve: the more engaged a student is, the more mindful her practice and the more time she spends on activities, which, in turn, leads to better outcomes [9, 46, 45]. We included both self-reported and trace-based (number of optional exercises completed) measures of engagement. We hypothesize that:

Hypothesis 1: Students will be more engaged with Ingenium than with the traditional text-only versions of the exercises.

In Ingenium, the blocks depict the grammatical constraints and possible grammatical roles of each word. Students are thus compelled to consider the grammatical connections among words, independent of word order, in order to build sentences with the blocks. We therefore expect that:

Hypothesis 2: Students will commit fewer errors related to word order when using Ingenium than using the traditional text-only versions of the exercises.

and

Hypothesis 3: Students will report that they reflected more on the grammar when using Ingenium than when using the traditional text-only versions of the exercises.

Ingenium explicitly encourages learners to focus their attention on the grammar of each sentence. While it makes the grammatical roles of the words particularly prominent, we expect that the overall attention on the grammar will also result in lower incidence of errors, unrelated to word order. For this reason, we expect that:

Hypothesis 4: Students will be able to complete translation exercises with fewer overall errors using Ingenium than using the traditional text-only versions of the exercises.

Ingenium makes examining grammar less intimidating and more intuitive. We therefore expect that:

Hypothesis 5: Students will report a greater positive change in self-efficacy related to Latin grammar and reading comprehension after using Ingenium than after using the traditional text-only versions of the exercises.

Self-efficacy, or task-related confidence, is an effective predictor of motivation [4, 48, 64].

Finally, because Ingenium is more visual, interactive, and hands-on, we expect that:

Hypothesis 6: Students will prefer using Ingenium over the traditional text-only versions of the exercises.

2. pecora ferit horror fessa

[noun] [verb] [noun] [adjective]

horror, horroris, m.: fear

Enter translation...

Figure 4: Sample sentence translation in the text-only condition. Parts of speech are indicated in dark gray beneath each word. The user here has right clicked to gloss, or to perform a dictionary lookup, of the highlighted word, *horror*, whose dictionary entry appears in light gray below.

Tasks

We designed two activities that are common in Latin instruction: translation exercises and multiple choice fill-in-the-blank exercises. In the control (text-only) condition, the exercises were presented in textual form (as in Figure 4). In the Ingenium condition, each sentence was additionally accompanied by a set of puzzle pieces. Otherwise, the control and the Ingenium conditions were identical in form and structure. The sentences in the two conditions were isomorphic: for each sentence in one condition there was an equivalent sentence in the other condition that used similar grammatical forms and interactions between words, and used vocabulary words at the same level of difficulty.

Sentences used for translation and fill-in-the-blank exercises were largely quoted or adapted for the novice level from commonly read Latin texts, including Virgil's *Aeneid* and *Georgics*, Cicero's *First Oration Against Catiline* and *Letters to Atticus*, Catullus' *Poem 64*, and Caesar's *Commentaries on the Gallic War*, among others.

In both conditions, students were presented with three rounds of exercises. Each round consisted of a set of four translation exercises and one fill-in-the-blank exercise. The successive experimental rounds increased in the complexity of interactions among words: 1) interactions between nouns and verbs; 2) interactions among nouns, verbs, and adjectives; 3) interactions among nouns, verbs, adjectives, adverbs and prepositions.

In the Ingenium condition, using Ingenium was required for the first two and optional for the last two translations in each round. In all fill-in-the-blank exercises, the use of Ingenium was always completely optional. For both conditions, students could skip the last round (the third set of translations and the third fill-in-the-blank exercise) at any point by pressing a button placed at the top right corner of their screen, which read: "I'm finished using this tool."

Participants

Of the 171 participants who navigated to the study home page, 67 participants completed the entire study. The youngest participants reported to be in the age range of 18–22 and the oldest reported being 51+ years old. Most were college students, who hailed from 42 different colleges and universities across the United States. 4 learners participated in person and 63 remotely. Participants included current and former students of Latin. Their formal study of Latin included representation from over 14 different Latin textbooks.

Participants were compensated $10 for taking part in the study (in-person participants received cash, remote participants Amazon gift certificates).

Procedures

In-person participants were recruited through visits to college Latin classes, and remote participants through email correspondence with Classics departments at over 50 colleges and universities across the United States, email notices to a local college's house lists and other internal lists, and through announcements on social media (Facebook and Twitter).

All the questionnaires and study activities were implemented for web-based use. All participants were asked to begin the study by navigating to the study home page on their computers.

Remote participants were screened through an online questionnaire that asked for their age, and level and years of Latin. Only adults with 1/2 year to 1 1/2 years of college or high school Latin were deemed eligible to participate and were given the link to the study.

At the start of the study, in-person participants were told that they should speak out if they found anything confusing, but that the experimenter could not answer their questions or interfere once the study began. All participants were then asked to fill out a questionnaire on their basic demographics and Latin experience. Next, they answered questions related to their self-efficacy, or confidence, with fundamental grammatical concepts in Latin.

Participants were then randomly assigned with a 50% probability to perform either the control or the experimental condition first. They were also randomly assigned with a 50% probability to either the inline or the external design of Ingenium in the experimental condition. Prior to starting each condition, participants were presented with condition-specific instructions. They then proceeded to the three rounds of exercises.

After completing each condition, participants were presented with several questions on a 7-point Likert scale. The questions pertained to self-reported cognitive and emotional engagement, level of attention paid to the grammar, and self-efficacy. The questions were identical in both conditions. After both conditions and post-assessments were completed, participants were asked which tool they preferred: the first or the second.

Adjustments of Data

Due to a programming error, 17 participants were able to complete the study without filling out the demographics questionnaire and the pre-assessment. Because these 17 participants still completed the required sets of exercises, they were only excluded from the self-efficacy analysis, which required pre-assessment data, but were included in the analysis of other measures.

Design and Data Analysis

This was a within-subjects study with one factor (exercise presentation: {text-only, Ingenium}) and the following measures (all subjective measures, except for the overall preference, were collected on a 7-point Likert scale):

- **Trace-based engagement.** In both conditions, as mentioned previously, all the exercises in the third round (4 translations and 1 fill-in-the-blank exercise) were optional. We used the number of optional exercises completed as a trace-based measure of engagement.

- **Self-reported measures of engagement.** After each condition, we asked participants two questions related to their cognitive engagement and two related to their emotional engagement. The answers to all four questions were averaged into a single measure of self-reported engagement prior to analysis.

- **Errors of word order.** For the 9 out of 12 translation exercises per condition that were not given in English word order, the first author flagged the incorrect translations that were caused by over-reliance on English word order. The assessor was blind to the condition in which the answer was generated while grading the accuracy of the answers.

- **Reflection on grammar.** After each condition, we asked participants how much they reflected on the grammar of the sentences during the activity that they just completed.

- **Overall error rate.** For all translation exercises completed by the participants, the first author assessed the accuracy of participants' answers. The assessor was blind to the condition in which the answer was generated while grading the accuracy of the answers.

- **Change in self-efficacy.** We asked participants five self-efficacy questions (related to Latin grammar and text comprehension) at three points during the study: at the beginning of the study and upon the completion of each condition. We computed the change in self-efficacy as the difference between self-efficacy just after and just before each of the two conditions. The answers to the five questions were averaged prior to analysis.

- **Overall preference.** At the end of the study, we asked participants to rank the two tools in the order of overall preference.

We used logistic regression (a generalized linear model with binomial distribution) to analyze trace-based engagement, for which the measures were binary (an optional exercise was either completed or not). We used participant ID as a co-variate in those analyses. For all other analyses, we used the Wilcoxon Signed Rank test because normal distribution could not be assumed in our data. We calculated Cohen's d for the effect size of errors related to word order and overall error rate.

To guard against Type I errors, we applied the Holm's sequentially-rejective Bonferroni procedure [24, 51] to the analyses. This method introduces fewer Type II errors than the more popular simple Bonferroni correction.

Results

Initial Analyses

We conducted an initial between-subjects analysis to see if the two designs of verb puzzle pieces (the *inline* versus the *external* design illustrated in Figure 2) resulted in different errors rates in translation exercises. Participants made fewer errors when using the external ($M = 43\%$) than the inline ($M = 46\%$) interfaces, but the difference was not statistically significant ($\chi^2_{(959.5, N=1)} = 0.4443, n.s.$). For that reason, in the subsequent analyses we combined results from the two variants of the Ingenium condition.

Main Analyses

The main results are summarized in Table 1.

Participants chose to complete more optional translation and fill-in-the-blank exercises when using Ingenium ($M = 59.1\%$) than when using the text-only interface ($M = 53.4\%$), and this difference was statistically significant ($\chi^2_{(1, N=670)} = 7.55, p = 0.006$). Participants also reported significantly higher levels of self-reported engagement with Ingenium ($M = 5.37$) than with the traditional interface ($M = 3.96$, $S = 643.00$, $p < 0.0001$). Hypothesis 1 is therefore supported.

Participants committed fewer errors related to word order using Ingenium ($M = 7.0\%$) than the text-only interface ($M = 20.0\%$); this effect was large (Cohen's $d = 0.82$) and statistically significant ($S = -386.00$, $p < 0.0001$). Hypothesis 2 was also supported.

Participants reported that they reflected on the grammar more after using Ingenium ($M = 5.54$) than after using the text-only interface ($M = 4.19$), and this difference was statistically significant ($S = 378.00$, $p < 0.0001$). Hypothesis 3 was thus supported.

Participants performed translation exercises with significantly lower overall error rates using Ingenium ($M = 45.1\%$) than using the text-only interface ($M = 57.4\%$); this effect size was moderate (Cohen's $d = 0.49$) and statistically significant ($S = -386.00$, $p < 0.0001$). Hypothesis 4 was supported.

Participants ($N = 50$) reported a positive change in self-efficacy after using Ingenium ($M = 0.196$), while their self-efficacy *decreased* slightly after using the text-only interface ($M = -0.072$). This difference was not statistically significant ($S = 139, n.s.$), however, so Hypothesis 5 was not supported.

Finally, participants preferred Ingenium (*mean rank* = 1.33) over the traditional interface (*mean rank* = 1.67), and this difference was statistically significant ($S = -391$, $p = 0.0042$). Hypothesis 6 was, therefore, supported.

DISCUSSION, FUTURE WORK & CONCLUSION

We developed Ingenium with two goals in mind: (1) to help students focus their attention on the grammar of the Latin sentences in order to aid them with correctly deriving their grammatical, and consequently semantic, compositions, and (2) to do so in a manner that made examining Latin grammar engaging.

Hypothesis		Text-only	Ingenium	Raw p-values	Adjusted p-values	
H1	Trace-based engagement	53.4%	59.1%	0.0060	0.0120	*
	Self-reported engagement	3.96	5.32	< 0.0001	< 0.0006	*
H2	Errors of word order	20.0%	7.0%	< 0.0001	< 0.0007	*
H3	Reflection on grammar	4.19	5.54	< 0.0001	< 0.0005	*
H4	Overall error rate	57.4%	45.1%	0.0010	0.0040	*
H5	Change in self-efficacy	−0.072	0.196	0.0602	0.0602	
H6	Overall preference	1.67	1.33	0.0042	0.0126	*

Table 1: Summary of the results. All subjective measures except the overall preference were reported on a 7-point Likert scale. Overall preference is reported as mean rank; lower indicates greater preference. We used the Holm's sequentially-rejective Bonferroni procedure to account for multiple hypotheses being tested simultaneously. We report both raw and adjusted p-values. Statistically significant results are marked with an asterisk.

In our study, students reported reflecting more on the grammar of the sentences when using Ingenium than when solving the traditional text-only versions of the exercises. This reinforces the main tenets of the Michigan Latin approach of focusing students' attention on grammatical concepts, rather than on word order and vocabulary [40]. This increased attention on the grammar coincided with improved reading comprehension performance: students translated their sentences more accurately when they could use Ingenium. In particular, they made much fewer errors related to word order when using Ingenium (7%) compared to a traditional text-only interface (20%). Other errors, mostly of tense (surrounds/surrounded) and number (animal/animals), were related to individual words in isolation, as opposed to interactions between words, and thus were not targeted by the design of Ingenium. These types of errors still decreased, though to a lesser extent. These findings support the success of the system in enabling students to decipher the meaning of Latin sentences by reflecting more on the grammar.

The results of our study also demonstrate that Ingenium can increase student engagement, both measured as the number of optional exercises they were willing to complete and as their self-reported assessments of emotional engagement and cognitive engagement. Students also indicated a strong preference for doing exercises with Ingenium over the traditional text-only versions. When the use of blocks in the translation exercises was optional, nearly all students still used them at least briefly (97%) and most completed the puzzle (79%).

In the future, we plan to develop ways of visualizing interactions and concepts of greater complexity, such as relative clauses, interrogative questions, clause subordination, etc. We also intend to explore collaborative mechanisms that would allow students to interact with Ingenium together. Because the restructuring of grammar commonly occurs in second language acquisition [42], Ingenium—while particularly fitting for Latin—can also be adapted to a wide variety of other morphologically rich languages, such as those in the Slavic and Finno-Ugric families.

We have already implemented a more open-ended sandbox interface, which was not deployed for the experiment, where students could create spontaneous sentences using novice-level vocabulary from the widely used Cambridge Latin Course, in addition to an interface that enables learners and instructors to input their own words that would generate Ingenium blocks from a lookup in the online Lewis and Short *Latin Dictionary*.[3] We prepare to investigate these more versatile and customizable interfaces of Ingenium with students and instructors.

To conclude, Ingenium addresses a long-standing challenge in Latin instruction: that of helping students make effective use of Latin grammar when comprehending the semantics of complex Latin texts. Unlike prior attempts to address this challenge, Ingenium uses accessible vocabulary of an intuitive visual language instead of idiosyncratic technical jargon. Interactions with Ingenium focus students attention on the process of grammatically unpacking a sentence as a means toward uncovering its meaning. Given its positive impact on students' engagement and performance, and its web-based implementation that allows it to be conveniently accessed online, we hope to develop Ingenium to a point where it becomes an easily adoptable, supplementary instructional tool for Latin students worldwide.

REFERENCES

1. Harshit Agrawal, Rishika Jain, Prabhat Kumar, and Pradeep Yammiyavar. 2014. FabCode: Visual Programming Environment for Digital Fabrication. In *Proceedings of the 2014 Conference on Interaction Design and Children (IDC '14)*. ACM, New York, NY, USA, 353–356. DOI: **http://dx.doi.org/10.1145/2593968.2610490**

2. Dominik Aufderheide, Werner Krybus, and Ulf Witkowski. 2012. Experiences with LEGO MINDSTORMS as an Embedded and Robotics Plattform within the Undergraduate Curriculum. In *Advances in Autonomous Robotics*, Guido Herrmann, Matthew Studley, Martin Pearson, Andrew Conn, Chris

---

[3]The Lewis and Short *Latin Dictionary* is a standard and widely used Latin dictionary that is available online through Tufts University's Perseus project at www.perseus.tufts.edu.

Melhuish, Mark Witkowski, Jong-Hwan Kim, and Prahlad Vadakkepat (Eds.). Lecture Notes in Computer Science, Vol. 7429. Springer Berlin Heidelberg, 185–196. DOI:`http://dx.doi.org/10.1007/978-3-642-32527-4_17`

3. Maurice Balme and James Morwood. 1997. *Oxford latin course. Part II.* Oxford University Press Oxford.

4. Albert Bandura. 1977. Self-efficacy: toward a unifying theory of behavioral change. *Psychological review* 84, 2 (1977), 191.

5. David Bau. 2015. Droplet, a Blocks-based Editor for Text Code. *J. Comput. Sci. Coll.* 30, 6 (June 2015), 138–144. `http://dl.acm.org/citation.cfm?id=2753024.2753052`

6. David Bau, D. Anthony Bau, Mathew Dawson, and C. Sydney Pickens. 2015. Pencil Code: Block Code for a Text World. In *Proceedings of the 14th International Conference on Interaction Design and Children (IDC '15)*. ACM, New York, NY, USA, 445–448. DOI:`http://dx.doi.org/10.1145/2771839.2771875`

7. Paolo Bottoni and Miguel Ceriani. 2015. Linked Data Queries As Jigsaw Puzzles: A Visual Interface for SPARQL Based on Blockly Library. In *Proceedings of the 11th Biannual Conference on Italian SIGCHI Chapter (CHItaly 2015)*. ACM, New York, NY, USA, 86–89. DOI:`http://dx.doi.org/10.1145/2808435.2808467`

8. Carrie J. Cai, Philip J. Guo, James R. Glass, and Robert C. Miller. 2015. Wait-Learning: Leveraging Wait Time for Second Language Education. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 3701–3710. DOI:`http://dx.doi.org/10.1145/2702123.2702267`

9. Robert M Carini, George D Kuh, and Stephen P Klein. 2006. Student engagement and student learning: Testing the linkages*. *Research in higher education* 47, 1 (2006), 1–32.

10. Catrobat. 2015. Pocket Code. `https://share.catrob.at/pocketcode`. (2015).

11. Yung-Ju Chang, Lezhong Li, Shih-Hsuan Chou, Min-Chih Liu, and Surong Ruan. 2013. Xpress: Crowdsourcing Native Speakers to Learn Colloquial Expressions in a Second Language. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 2555–2560. DOI:`http://dx.doi.org/10.1145/2468356.2468829`

12. Edith Frances Claflin. 1943. Teaching the Art of Reading Latin. *Classical Journal* (1943), 130–136.

13. Thomas M Connolly, Elizabeth A Boyle, Ewan MacArthur, Thomas Hainey, and James M Boyle. 2012. A systematic literature review of empirical evidence on computer games and serious games. *Computers & Education* 59, 2 (2012), 661–686.

14. Stephen Cooper. 2010. The Design of Alice. *Trans. Comput. Educ.* 10, 4, Article 15 (Nov. 2010), 16 pages. DOI:`http://dx.doi.org/10.1145/1868358.1868362`

15. S Erhel and E Jamet. 2013. Digital game-based learning: Impact of instructions and feedback on motivation and learning effectiveness. *Computers & Education* 67 (2013), 156–167.

16. Google. 2015. Google Blockly. `https://developers.google.com/blockly`. (2015).

17. Lindsay Grace, Martha Castaneda, and Jeannie Ducher. 2012. User Testing of a Language Learning Game for Mandarin Chinese. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems (CHI EA '12)*. ACM, New York, NY, USA, 865–868. DOI:`http://dx.doi.org/10.1145/2212776.2212859`

18. John Gruber-Miller, American Philological Association, and others. 2006. *When dead tongues speak: Teaching beginning Greek and Latin*. Oxford University Press.

19. Mark Guzdial. 2008. Education Paving the way for computational thinking. *Commun. ACM* 51, 8 (2008), 25–27.

20. William Gardner Hale. 1887. *The Art of Reading Latin: How to Teach It*. Ginn & Company.

21. Cheng-Hsien Han, Chi-Lan Yang, and Hao-Chuan Wang. 2014. Supporting Second Language Reading with Picture Note-taking. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems (CHI EA '14)*. ACM, New York, NY, USA, 2245–2250. DOI:`http://dx.doi.org/10.1145/2559206.2581204`

22. Brian Harvey, Daniel D. Garcia, Tiffany Barnes, Nathaniel Titterton, Daniel Armendariz, Luke Segars, Eugene Lemon, Sean Morris, and Josh Paley. 2013. SNAP! (Build Your Own Blocks) (Abstract Only). In *Proceeding of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE '13)*. ACM, New York, NY, USA, 759–759. DOI:`http://dx.doi.org/10.1145/2445196.2445507`

23. Brian Harvey and Jens Mönig. 2010. Bringing "No ceiling" to scratch: can one language serve kids and computer scientists. *Proc. Constructionism* (2010).

24. S. Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics* 6, 65-70 (1979), 1979.

25. Hopscotch. 2015. Hopscotch. `https://www.gethopscotch.com`. (2015).

26. E Hutchins, J Hollan, and D Norman. 1985. Direct manipulation interfaces. *Human-Computer Interaction* (1985).

27. Eleanor Huzar. 1957. Structural Linguistics and Latin Teaching. *Classical Journal* (1957), 268–274.

28. Charles Jenney, Rogers V Scudder, and Eric C Baade. 1987. *Jenney's First Year Latin*. Allyn and Bacon.

29. James F Johnson. 1987. Alternative Approaches for the College Elementary Latin Sequence. *Classical Journal* (1987), 246–255.

30. Caitlin Kelleher, Randy Pausch, and Sara Kiesler. 2007. Storytelling Alice Motivates Middle School Girls to Learn Computer Programming. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 1455–1464. DOI: `http://dx.doi.org/10.1145/1240624.1240844`

31. Glenn M Knudsvig. 1986. *Latin for Reading: A Beginner's Textbook with Exercises*. University of Michigan Press.

32. Geza Kovacs. 2013. Smart Subtitles for Language Learning. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 2719–2724. DOI: `http://dx.doi.org/10.1145/2468356.2479499`

33. Jerry Lee Jr and others. 2009. *Scratch programming for teens*. Cengage Learning.

34. Sorin Lerner, Stephen R. Foster, and William G. Griswold. 2015. Polymorphic Blocks: Formalism-Inspired UI for Structured Connectors. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 3063–3072. DOI: `http://dx.doi.org/10.1145/2702123.2702302`

35. Bob Lister. 2009. Latin in transition. *Arts and Humanities in Higher Education* 8, 2 (2009), 191–200.

36. Jiangjiang Liu, Cheng-Hsien Lin, Joshua Wilson, David Hemmenway, Ethan Hasson, Zebulun Barnett, and Yingbo Xu. 2014. Making Games a "Snap" with Stencyl: A Summer Computing Workshop for K-12 Teachers. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education (SIGCSE '14)*. ACM, New York, NY, USA, 169–174. DOI: `http://dx.doi.org/10.1145/2538862.2538978`

37. David J Malan and Henry H Leitner. 2007. Scratch for budding computer scientists. In *ACM SIGCSE Bulletin*, Vol. 39. ACM, 223–227.

38. John Maloney, Mitchel Resnick, Natalie Rusk, Brian Silverman, and Evelyn Eastmond. 2010. The scratch programming language and environment. *ACM Transactions on Computing Education (TOCE)* 10, 4 (2010), 16.

39. John H Maloney, Kylie Peppler, Yasmin Kafai, Mitchel Resnick, and Natalie Rusk. 2008. Programming by choice: urban youth learning programming with scratch. *ACM SIGCSE Bulletin* 40, 1 (2008), 367–371.

40. Donka D Markus and Deborah Pennell Ross. 2004. Reading Proficiency in latin through expectations and visualization. *Classical world* 98, 1 (2004), 79–93.

41. Daniel V McCaffrey. 2006. Reading Latin Efficiently and the Need for Cognitive Strategies. *When Dead Tongues Speak: Teaching Beginning Greek and Latin: Teaching Beginning Greek and Latin* 6 (2006), 113.

42. Barry McLaughlin. 1990. Restructuring. *Applied Linguistics* 11, 2 (1990), 113–128. DOI: `http://dx.doi.org/10.1093/applin/11.2.113`

43. Mitchel Resnick. 2007. All I really need to know (about creative thinking) I learned (by studying how children learn) in kindergarten. In *Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition*. ACM, 1–6.

44. Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, and others. 2009. Scratch: programming for all. *Commun. ACM* 52, 11 (2009), 60–67.

45. Timothy Rodgers. 2008. Student engagement in the e-learning process and the impact on their grades. *International Journal of Cyber Society and Education* 1, 2 (2008), 143–156.

46. Timothy Rodgers and Deb Ghosh. 2001. Measuring the determinants of quality in UK higher education: a multinomial logit approach. *Quality Assurance in Education* 9, 3 (2001), 121–126.

47. Krishnendu Roy. 2012. App Inventor for Android: Report from a Summer Camp. In *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education (SIGCSE '12)*. ACM, New York, NY, USA, 283–288. DOI: `http://dx.doi.org/10.1145/2157136.2157222`

48. Dale H Schunk. 1990. Goal setting and self-efficacy during self-regulated learning. *Educational psychologist* 25, 1 (1990), 71–86.

49. GM SELIGSON. 1983. LATIN AT MICHIGAN 1951-1981. In *Latin Linguistics and Linguistic Theory: Proceedings of the 1st International Colloquium on Latin Linguistics, Amsterdam, April 1981*, Vol. 12. John Benjamins Publishing, 295.

50. Gerda M Seligson. 1964. The Structural Approach, the First Decade. *The Classical World* (1964), 97–100.

51. Juliet P. Shaffer. 1995. Multiple Hypothesis-Testing. *Annual Review of Psychology* 46 (1995), 561–584.

52. Jonah WD Skiles. 1943. The Teaching of the Reading of Latin in the Latin Word-Order. *Classical Journal* (1943), 88–104.

53. Squeakland. 2015. Etoys. `http://www.squeakland.org`. (2015).

54. Waldo E Sweet. 1957. *Latin: a structural approach*. University of Michigan Press.

55. Waldo E Sweet. 1967. The Continued Development of the Structural Approach. *Didaskalos* 2, 2 (1967), 141–159.

56. V.L. Takacs. 2014. BlockImpress. In *Intelligent Engineering Systems (INES), 2014 18th International Conference on*. 221–226. DOI: `http://dx.doi.org/10.1109/INES.2014.6909372`

57. Feng Tian, Fei Lv, Jingtao Wang, Hongan Wang, Wencan Luo, Matthew Kam, Vidya Setlur, Guozhong Dai, and John Canny. 2010. Let's Play Chinese Characters: Mobile Learning Approaches via Culturally Inspired Group Games. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 1603–1612.

58. Jake Trower and Jeff Gray. 2015. Blockly Language Creation and Applications: Visual Programming for Media Computation and Bluetooth Robotics Control. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15)*. ACM, New York, NY, USA, 5–5. DOI: `http://dx.doi.org/10.1145/2676723.2691871`

59. Kevin Wang, Corey McCaffrey, Daniel Wendel, and Eric Klopfer. 2006. 3D Game Design with Programming Blocks in StarLogo TNG. In *Proceedings of the 7th International Conference on Learning Sciences (ICLS '06)*. International Society of the Learning Sciences, 1008–1009. `http://dl.acm.org/citation.cfm?id=1150034.1150223`

60. David Weintrop and Uri Wilensky. 2013. Robobuilder: A Computational Thinking Game (Abstract Only). In *Proceeding of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE '13)*. ACM, New York, NY, USA, 736–736. DOI: `http://dx.doi.org/10.1145/2445196.2445430`

61. Frederic M Wheelock. 1963. *Latin: An introductory course based on ancient authors*. Number 104. Barnes & Noble Imports.

62. Frederic M Wheelock and Richard A LaFleur. 2005. *Wheelock's Latin*. HarperResource.

63. Ursula Wolz, Henry H Leitner, David J Malan, and John Maloney. 2009. Starting with scratch in CS 1. In *ACM SIGCSE Bulletin*, Vol. 41. ACM, 2–3.

64. Barry J Zimmerman, Albert Bandura, and Manuel Martinez-Pons. 1992. Self-motivation for academic attainment: The role of self-efficacy beliefs and personal goal setting. *American educational research journal* 29, 3 (1992), 663–676.