Acta Universitatis Sapientiae, Philologica, **16**, 2 (2024) 68–87

DOI: 10.47745/ausp-2024-0017

Grammar Practices in the Digital World

Mária CSERNOCH
University of Debrecen (Debrecen, Hungary)
Faculty of Informatics
csernoch.maria@inf.unideb.hu
https://orcid.org/0000-0002-7088-7714

Attila IMRE
Sapientia Hungarian University of Transylvania (Cluj Napoca, Romania)
Department of Applied Linguistics
attilaimre@ms.sapientia.ro
https://orcid.org/0000-0003-3886-7091

Abstract. Teaching and protecting Hungarian language and cultural values have become more and more challenging due to the fast spread of uncontrollable digital platforms. In their text-based digital products (e-mail, chat, blog, etc.), youngsters prefer elliptical expressions (digital slang) to grammatically correct sentences. Schools, course books offer a wide variety of opportunities to practise grammar, but students find these exercises rather boring and consider them as school chores. As students live in the digital world, teachers should consider alternatives to offering exclusively classical, paper-based exercises.

The present work provides a subject-integrated approach, where paper-based course book tasks are converted into data management problems to practise grammar. This novel approach may enable students to become more engaged not only in solving the original grammar problems but also in finding digital solutions. Building algorithms helps them understand grammar rules, as well as differences between handwritten and computer-stored characters. It is also stipulated that our approach is open to generalization and suitable to solve similar problems in languages other than Hungarian.

Keywords: language, culture, content, grammar course books, digitization

1. Introduction

1.1. Digital pull system in education

This study is part of a longitudinal project to introduce a digital pull system that can replace the widely used but ineffective and inefficient push systems (Ohno 1988) in digital education. In digital push systems, the focus is on tools, including hardware and software resources, and thus software interfaces are taught in the hope to build up knowledge inventory, resulting in millions of erroneous digital documents, piling up waste. However, a pull system focuses on teachers and students by developing their computational thinking skills and problem-solving abilities. These values are supported by two principles:

– just-in-time, which means that raw material (data in our case) should arrive only by the time the production begins (the task requires);

– autonomation, which is "automation with a human touch" (Ohno 1988, Krafcik 1988, Liker 2004, Rother 2010, Womack and Jones 2003).

In the case of digital education, just-in-time refers to the fact that when problems are presented in classes, only the tools needed to solve the problems are introduced, and thus the focus is on the output which dictates the resources in demand. With this approach, no waste is produced (hence the pull system), and both teachers and students may learn how to work effectively and efficiently in digital education.

Along with the classical statistical education research methods, action research (Brydon-Miller et al. 2003, Eilks 2018) plays a crucial role in the project. This includes the teachers' task to find those problems/tasks which serve the students' best interest, considering both content and digital development, and to introduce these tasks during the classes. To support this approach, all the aspects of Technological Pedagogical Content Knowledge (TPCK) (Angeli and Valadines 2015) must be taken into consideration.

The second principle, autonomation, means to find the proper balance between automation and human intervention. If automation is efficient, then the process alone should suffice, but when any deviation from standards is discovered, the process must be stopped to correct the error(s).

In these circumstances, classroom situations are to be offered and evaluated by the teachers, who decide whether problems can be solved more effectively and efficiently with the classical methods, or digitization may add values to the teaching-learning process. In our approach, a crucial role is played by data analysis and repetition. As the presented solutions indicate, a thorough data analysis is required during digital data manipulation, as human intelligence is needed to explain the "dumb" computer how to carry out error-free automated or semi-automated solutions. However, repetition is unavoidable during the analyses while setting up the kanban,[1] building the algorithm, completing the coding, and checking the correctness of outputs. This method aims at repeating the knowledge pieces so many times in different contexts that it would ultimately build up schemata in the long-term memory (Kahneman 2011, Sweller 2011, Ghazi-Saidi and Ansaldo 2017).

1.2. The content of the paper

The present work summarizes a grammar-focused action research,[2] where paper-based course book tasks are converted into data management problems to practise language, develop students' computational thinking skills and digital problem-solving abilities. Furthermore, gains by digitization of these tasks are also discussed.

1.3. The aims of the paper

The main aim of the paper is to draw attention to the opportunities that lie in digitization. In educational contexts, digitization is the conversion process during which digital solutions are offered to paper-based tasks. The secondary aim is to show various solutions and tools which prove that considering all aspects of TPCK (Angeli and Valadines 2015) would improve students' content knowledge, digital skills, and problem-solving abilities.

2. Action research activities

2.1. Participating schools and teachers

In 2020, we launched the Digital School project in a primary school (grades 1–8), with the core aim to introduce digital solutions in non-informatics school subjects taking the form of computer-aided teaching and learning, in the frame of subject integration. In this project, led by a mentor, the focus is on the equilibrium of all aspects of TPCK (Mishra and Koehler 2006, Angel and Valanides 2015) during informatics and non-informatics classes. In non-informatics classes, novel, digital solutions can be introduced compared to the classical, paper-based solutions, as detailed in the present paper. In informatics classes, we aim at providing real-world content to teach computer sciences/informatics to leave behind the widely accepted non-contextualized, tool-centred approaches and methods (push systems).

1 The term refers to a tool of a pull production system which can manage and keep track of work.
2 Research method intended to both investigate and solve a perceived issue.

In the first year, the teachers of the Digital School project were engaged by learning fundamental informatics through real-world problems from various school subjects. The teaching-learning process took place in the computer lab of the school. In this tuition phase, along with the technical and content knowledge, the pedagogical and didactical aspects of this novel approach were introduced by applying kaizen (Maurer 2013, 2014). In this context, kaizen includes the introduction of the tiny changes which bypass our mental alarm system, allowing our creative and intellectual processes to flow without obstruction. The result is a lasting and powerful change (Maurer 2013, 2014). The reason for the presentation of the little steps of kaizen is to show teachers how they should handle problems in classes in a digital environment. Even digital students need as much time as teachers to make fundamental changes in their behaviour, in this case switching from the deeply coded and secure paper-based environment to creative and productive digital solutions (Kirschner and De Bruyckere 2017, Kirschner et al. 2006).

Furthermore, one of the most crucial aspects of the Digital School project is that the tuition of the teachers did not finish at the end of the first year. It still goes on under various forms:

– the mentor visits and supervises the digital classes;

– the mentor gives lessons recorded by the teachers;

– the teachers participate in small-group lessons and discussions to learn and tune the method;

– the teachers prepare digital course plans with the help of the mentor;

– the teachers go to post-graduate courses offered by the University of Debrecen;

– each year, the school organizes an open day to present examples of how a digital school advances with the help of a dedicated mentor.

In this school, informatics is taught from grade 1, one class per week for everyone, which is widely accepted throughout Hungary despite that informatics (Digital Culture) is compulsory from grade 3 (NAT2020 2017).

At the time of the preparation of the present paper, the project is in the phase of action research (Groundwater-Smith and Campbell 2010). The teachers were interviewed, and their classes were regularly visited and discussed. These classes were also presented during the open days, and those who participated at the university courses summarized the results in their dissertations (Bényei 2023, Szűcs 2023, Ékesné 2023, Gyarmati 2022, Kormány 2022, Kamarás 2022).

On the other hand, there is another school, which is typically not open to digital changes. However, we found a pre-service teacher with linguistics background and a grammar teacher in this school who were open to try out the method in their classes, and the third participant is the mentor who works as a substitute teacher in this school.

In the second school, one of the groups studied informatics in three classes a week with the method of the digital school, taught by the mentor. Their experience included subject integration and the application of kaizen and enquiry-based learning while studying informatics. The other group in the second school studied regular informatics in only one class a week. None of these groups used computers in any other subject except informatics.

2.2. Missing competence

We found that in classes where students do not do these subject integration exercises regularly – late starter classes in the digital school and the normal class in the second school –, more problems are met considering digital skills than those who take part in the Digital School project taught by "digital" teachers (their digital background is detailed above) or the mentor. The most frequent problems are the following:

– shortcomings in analytical and integrational skills;
– selecting the proper software to solve a problem;
– handling files (e.g. naming files properly, converting files, creating folders, finding folders);
– uploading files to the named online classroom and assessment slot;
– creating and formatting tables (e.g. setting up the properties, deciding on the number of fields (columns), organizing content into meaningful records, handling borders, handling the alignments of contents);
– handling figures, where the most serious problem is to decide whether pixel or vector graphics applications are needed to solve a problem.

These problems proved to be so demanding that it did not allow students to focus on the grammar task; consequently, teachers were obliged to teach fundamental informatics, thus leading to a failure in achieving the primary task, which was ultimately carried out during the next or the next two classes.

2.3. Preparation for digital classes

Considering hardware tools in the Digital School project, in grade 1 primarily digital boards are used by both teachers and students (where they have access to such a tool). From grade 2 up, students have their digital tools: in grades 2 and 3, tablets and from grade 4 laptops. On all the tools, folders are set up for each student and within their folders, subfolders for the different subjects. The course and exercise books are preloaded to these folders through the local net of the school. Students can save their solutions in these folders, which includes that they have meaningful folder- and file-handling exercises from a very early age. This result seems extremely crucial, since students lacking this skill struggle even with handling files (Csernoch 2021).

2.4. Skills gained

As the action research activities present, the participating teachers have regular meetings and various other forums to share their experience and develop their methods and the approach in general. The behaviour and reflection of the students reveal that the selection of the tasks plays a crucial role in their development, which explains the focus of the present paper. It has been found that the tasks suitable for digitalization mostly focus on the content instead of the tools. This finding includes that digital solutions allow space for deeper analysis of the tasks and data, faster implementation, and wider discussion. Furthermore, we must mention the motivation aspect of these digital solutions. Students are delighted to use digital tools to become designers of something valuable and be able to work fast with tools they are familiar with and make them part of their everyday life.

The approach has a great impact on the students struggling with learning disabilities. The combination of kaizen (Maurer 2013, 2014) and the enquiry-based teaching-learning (Polya 1945) method unlocks most of their barriers. It has been revealed that these students have common intelligence and abilities (e.g. in the case of dysgraphia, the lack of fine motor skills hinders them in their development), while the lack of coherence can be often detected, a shortcoming that can be improved through this approach.

3. Digital tasks

Saving a file for the first time with a proper given name is the primary action to be carried out in the case of each digital problem, followed by repeated savings. Students should learn how to name files properly: a brief reference to the content and the author with only the 26 letters of the English alphabet, completed with the underscore character between the words to avoid whitespace. Once teachers demand it consistently, file naming becomes a routine activity (schema, standard) that can be activated with fast thinking in various situations (Kahneman 2011). In the following, opening and saving files will not be listed as activities.

3.1. Conditional colouring

A well-known colouring task for grade 2 is presented in *Figure 1* (Fülöp and Szilágyi 2021: 7), during which children are asked to colour the consonants in blue and the vowels in red. This kind of task is quite popular among course book authors; consequently, it can be found in any course and exercise book with a variety of content. On the other hand, both children and parents find these tasks boring and time consuming compared to their values.

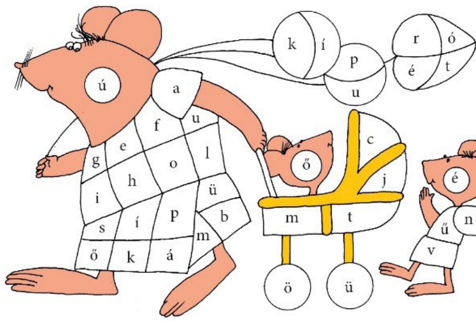

Figure 1. *Grade 2 grammar book task (Fülöp and Szilágyi 2021: 7)*

3.1.1. Colouring: Paper-based solution

In its original form, the task focuses more on the colouring details rather than on the content. At the age of seven, students have difficulties with colouring inside the lines, they mix up pencils, and there is no way for proper correction (erasing might smear the picture and/or make holes in the paper, etc.), as illustrated in *Figure 2*. Generally speaking, correction is a nightmare, and completing the task takes a long time, without focusing on the original vowel-consonant issue.

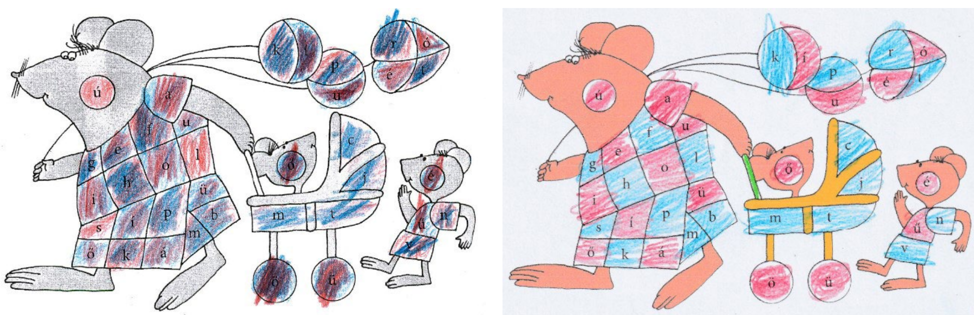

Figure 2. *Two classical paper-pencil colouring solutions for the grade 2 task in* Figure 1

3.1.2. Colouring: Digital solution

In an image manipulation program (GIMP is a good choice), students can select a pen colour (*Foreground Colour*) and fill in the areas defined by the borders with the *Bucket Fill Tool* (*Figure 3*).

Changing the colour is simple, which incorporates the easy correction of mixed-up letters/shapes and the colouring of small details (*Figure 3*). The use of Ctrl + scrolling (modification key with mouse) or the selection from a dropdown list can be introduced to zoom in/out the image. In this solution, the focus is on the original task, since the colouring tool is simple. Furthermore, students can set up their own algorithms considering content, time, punctuality, efficiency, etc. If the teacher is more “adventurous”, RGB can be introduced to select pure red and blue: RGB (255,0,0) and RGB (0,0,255) respectively.

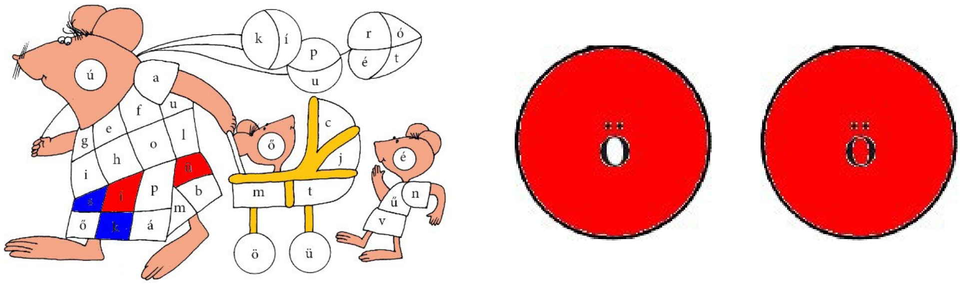

Figure 3. *The colouring process (left) and the letter ‘ö’ zoomed in to fill in small pieces in GIMP (task in* Figure 1*)*

A further advantage of solving this task in GIMP is that it provides opportunity to practise mouse actions – clicking with the left button, scrolling, and scrolling with modification key.

3.2. Separating words

The main task is to split a string into words[3] (*Figure 5*). A possible literal translation is *whatmeansoftransportdoyouusetogettoschool*. Then Part a) asks to write the sentence correctly, reminding the students the sentence-start capitalization and the sentence-end punctuation mark. Part b) of the task is to answer the question and write down the sentence (Fülöp and Szilágyi 2022: 7) (*Figure 5*).

temilyenközlekedésieszközzeljutszelaziskoláig

Tagold a következő mondatot szavakra!

temilyenközlekedésieszközzeljutszelaziskoláig

a) Írd le helyesen a mondatot! Ne feledkezz meg a mondatkezdő betűről és a mondatvégi írásjelről!

b) Válaszolj a leírt kérdésre! Írd le a mondatodat!

Figure 5. *Split the sentence into words (Fülöp and Szilágyi 2022: 7)*

3 The task probably entails that only meaningful Hungarian words should be looked for, and the entire string of letters should be used to carry it out successfully.

3.2.1. Separating words: Paper-based solution

In the original printed form, the string may be separated with vertical lines (word boundaries) (*Figure 6*), and then students can write the correct sentence in the two empty lines in Part a) (*Figure 5*).

Figure 6. *Splitting the sentence into words with vertical lines (Fülöp and Szilágyi 2022: 7)*

3.2.2. Separating words: Digital solution

The step-by-step digital solution to this task is presented in *Figure 7*. As the student advances in the string (from left to right), the Hungarian-language proofing tool recognizes the words separated from each other with spaces and gradually removes the red underline. Finally, the first letter should be capitalized and the proper punctuation mark added to close the sentence. Capitalization of sentence-start word and sentence-end punctuation mark allows us to introduce the *Home* and *End* keys, which instantly move the cursor to the beginning and end of the current line.

temilyenközlekedésieszközzeljutszelaziskoláig¶

te·milyenközlekedésieszközzeljutszelaziskoláig¶

te·milyen·közlekedésieszközzeljutszelaziskoláig¶

te·milyen·közlekedési·eszközzeljutszelaziskoláig¶

te·milyen·közlekedési·eszközzel·jutszelaziskoláig¶

te·milyen·közlekedési·eszközzel·jutsz·elaziskoláig¶

te·milyen·közlekedési·eszközzel·jutsz·el·aziskoláig¶

te·milyen·közlekedési·eszközzel·jutsz·el·az·iskoláig¶

Te·milyen·közlekedési·eszközzel·jutsz·el·az·iskoláig?¶

Figure 7. *The step-by-step digital solution of the word separation task presented in* Figure 5

The following knowledge pieces are required to complete the task in MS Word:
– text copied without formatting into MS Word,
– changing font type to Times New Roman and font size to 16 pt in the *Normal* style,

– moving the cursor with the right (optionally the left) direction key,

– typing the Space character (the non-printable Space character is displayed on the screen as a small dot in mid-height),

– moving the cursor to the beginning of the line (*Home* key),

– deleting the first letter of the sentence with the *Delete* key (since the letter is on the right side of the cursor),

– inserting an uppercase letter by holding down the *Shift* key and typing *t* (Shift + T),

– moving the cursor to the end of the line (*End* key),

– typing the "?" character.[4]

While the first two steps (copying without formatting and setting the *Normal* style) have already been applied before, only the subsequent steps are new in the series.

3.3. Missing accents

Many languages contain language-specific characters or so-called accents (Bringhurst 2004: 89) accompanying the standard 26 English letters. These accents abound in Hungarian, and words with or without accents can either drastically change the meaning of the words or make them meaningless. For instance, acute accent is present in the Hungarian – *á, é*, *í*, *ó*, and *ú* –, but there is double acute (*ő* and *ű*), or umlaut/diaeresis as well (*ö* and *ü*). Consequently, Hungarian teachers must pay attention to teach how to type words containing accents correctly. One of these exercises is presented in *Figure 14* (Fülöp and Szilágyi 2022: 9). The task explains that the words were printed without any accents, and students are asked to apply the missing ones and then read the completed words with the correct pronunciation. After checking and correcting them, they are asked to copy the words in their exercise books.

repulogep, papır, kerıtes, gyonyoru, mındıg, gyogyıt,

kapuja, pokhalo, huszonot, fesu, borond, porszıvo

Figure 14. *Words without proper Hungarian accents (Fülöp and Szilágyi 2022: 9)*

3.3.1. Missing accents: Paper-based solution

In the printed version, students should place (draw) the missing accents above the characters (*Figure 15*).

4 While the standard US keyboard layout offers the question mark on the left side of the right *Shift* key, the Hungarian keyboard layout has this mark to the right side of the *M* key (Shift + ,).

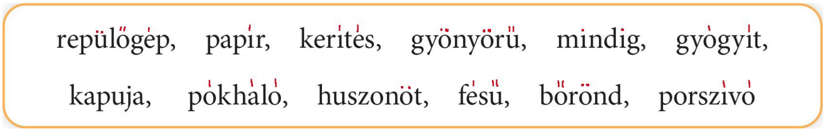

Figure 15. *Placing the missing accents in the Hungarian words (Figure 14)*

3.3.2. Missing accents: Digital solution

In the digital solution, students are asked to change the incorrect characters to the correct ones. The task may be carried out in two conceptually different ways.

– deleting the incorrect character and then inserting the correct one

- the *Delete* or the *Backspace* key deletes the wrong character if it is on the right or left side of the cursor respectively

– overwriting the incorrect character

- with selection or
- in the overwrite mode of operation of the keyboard

To solve this problem, the most effective solution is the use of the overwrite mode of the keyboard. However, it depends on the teachers whether they encourage their students to try any of the above-listed solutions. The status bar in *Figure 16* shows that the actual keyboard mode is overtype.

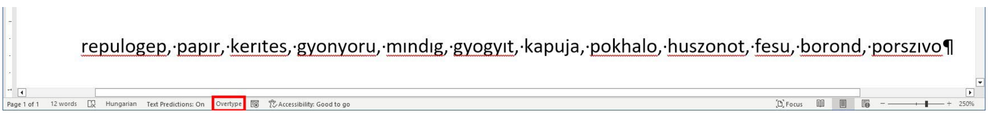

Figure 16. *Words without accents (*Figure 14*) with the overwrite mode of the keyboard shown on the status bar*

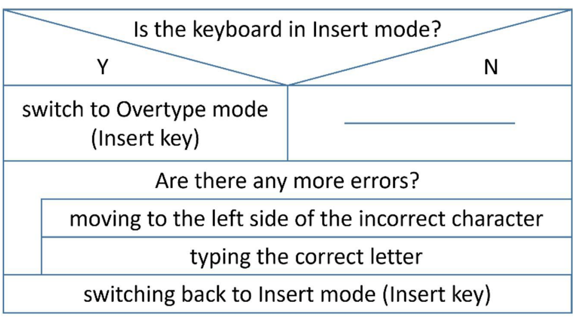

Figure 17. *The algorithm of overwriting the incorrect characters in* Figure 14

The students should carry out the following algorithm to correct the characters with the overwrite mode of the keyboard (*Figure 17*):

– text copied without formatting into MS Word,

– deleting the paragraph mark at the end of the first line (moving the cursor to the end of the line (*End* key) and then pressing the *Delete* key),

– switching to overwrite mode by pressing the *Insert* key once if the keyboard mode is insertion (*Insert* shown on the status bar),

– positioning the cursor on the left side of the incorrect character and then typing the correct letter.

3.4. Spelling

Spelling is of particular interest in teaching any (foreign) language. Compound words (written as one word, hyphenated, or in two separate words) is an acute issue in Hungarian, especially with geographical names. This is well reflected in a grade 6 exercise book (Sápiné Bényei 2021: 16–17), asking the students to spell them correctly, by writing one of the three options (1 – one word, 2 – separated, x – hyphenated) next to the words, resembling a betting (*Figure 18*). In the digital solution, the task can be converted into an Excel workbook (*Figure 19*).

Vas + megye		
Szentendrei + sziget		
Balaton + füred		
Fekete + tenger		
Erzsébet + híd		
Dobó István + tér		
Atlanti + óceán		
János + hegy		
Lengyel + Köztársaság		
Északi + középhegység		
Nagy + Kanizsa		
Cserhát + hegység		
Angyal + föld		
Új + Szeged		

Figure 18. *A spelling task from a grade 6 exercise book (Sápiné Bényei 2021: 16–17)*

3.4.1. Spelling: Paper-based solution

To solve the problem on paper and clarify whether there is a space between the two words, an extra column is set up to write 1, 2, or x. After deciding the separator, students should write the correct names in the third column (*Figure 18*).

3.4.2. Spelling: Digital solution

The original task can be converted to a spreadsheet workbook, where the following algorithm presents one possible solution:

– carrying out the conversion process with formulae (*Figure 19* Columns B and D, Fields *first* and *second*);

– solving the task by typing the required character or leaving the space empty (*Figure 19* Column D, Field *typing*);

– setting up conditional formatting to clearly indicate the space character and the empty cell (*Figure 19* Column D, Field *typing*);

– writing a formula to concatenate the strings (*Figure 19* Column E, Field *solution*);

– correcting the formula to get the proper solution (*Figure 19* Columns F and G, Fields *correction* and *correct_solution*).

For each step, a separate worksheet is recommended. *Figure 19* shows the final sheet with all the consecutive steps and the name of the previous sheets (*original*, *separating*, *typing*, *solution1*, *correction*, and *solution2*).

	A	B	C	D	E	F	G
1	original	first	second	typing	solution	correction	correct_solution
2	Vas + megye	Vas	megye		Vas megye	megye	Vas megye
3	Szentendrei + sziget	Szentendrei	sziget	-	Szentendrei-sziget	sziget	Szentendrei-sziget
4	Balaton + füred	Balaton	füred		Balatonfüred	füred	Balatonfüred
5	Fekete + tenger	Fekete	tenger	-	Fekete-tenger	tenger	Fekete-tenger
6	Erzsébet + híd	Erzsébet	híd	-	Erzsébet-híd	híd	Erzsébet-híd
7	Dobó István + tér	Dobó István	tér		Dobó István tér	tér	Dobó István tér
8	Atlanti + óceán	Atlanti	óceán	-	Atlanti-óceán	óceán	Atlanti-óceán
9	János + hegy	János	hegy	-	János-hegy	hegy	János-hegy
10	Lengyel + Köztársaság	Lengyel	Köztársaság		Lengyel Köztársaság	Köztársaság	Lengyel Köztársaság
11	Északi + középhegység	Északi	középhegység	-	Északi-középhegység	középhegység	Északi-középhegység
12	Nagy + Kanizsa	Nagy	Kanizsa		NagyKanizsa	kanizsa	Nagykanizsa
13	Cserhát + hegység	Cserhát	hegység	-	Cserhát-hegység	hegység	Cserhát-hegység
14	Angyal + föld	Angyal	föld		Angyalföld	föld	Angyalföld
15	Új + Szeged	Új	Szeged		ÚjSzeged	szeged	Újszeged

original | separating | typing | solution1 | correction | solution2

Figure 19. *A spreadsheet solution of the spelling task (Sápiné Bényei 2021: 16–17)*

The field *original* is created with an unformatted copying from the original exercise book, as it is discussed in the previous tasks (*Figure 19* Column A).

The values in the field *first* are calculated with a formula. In this field, the first string of the original expression (the characters on the left side of the + sign) should be displayed. Here, a backward method is applied, called pull system in industry. The lost action goes back and requests the previous one, while the second lost goes back to its previous, etc. (Ohno 1988, Krafcik 1988, Liker 2004, Rother 2010, Womack and Jones 2003).

– before displaying the first string, its length must be known;

– calculating the length of the first string requires the position of the + sign;

– a linear search algorithm is needed to locate the position of the + sign.

When all this is clear, the algorithm is set up in a forward order (*Figure 20*), followed by the creation of the formulae (*Table 1*).

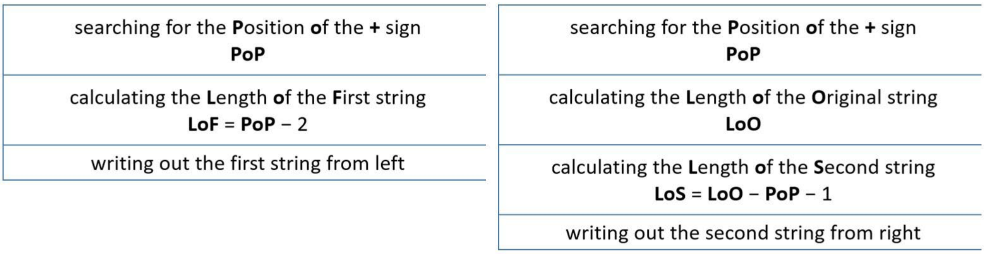

Figure 20. *The algorithms of separating the first (left) and the second (right) strings*

In this paper, all the given formulae are array formulae, indicated by the brackets around the formulae and vectors as input values (Csernoch 2014; Sestoft 2011; Microsoft Support 2023; Csapó et al. 2019, 2020) detailed in *Tables 1–5*. Thus, we can avoid copying the formulae, which is one of the most erroneous tools in spreadsheeting (Panko 1998, 2012).

Table 1. *The development of the formula to write out the first string from the original expression*

PoP	{=SEARCH(“+”,A2:A15)}
LoF	{=SEARCH(“+”,A2:A15)-2}
first string	{=LEFT(A2:A15,SEARCH(“+”,A2:A15)-2)}

Similarly to the procedure set up for the first string, we can decide what we need to write out in the second string, and then we can create the algorithm (*Figure 20*).

– before displaying the second string, its length must be known;

– calculating the length of the second string requires the position of the + sign and the length of the original string (this length requires the number of characters in the string);

– a linear search algorithm is needed to locate the position of the + sign.

Based on the algorithm, we can do the coding to write out the second string (*Table 2*).

Table 2. *The development of the formula to write out the second string from the original expression*

PoP	{=SEARCH("+",A2:A15)}
LoO	{=LEN(A2:A15)}
LoS	{=LEN(A2:A15)-SEARCH("+",A2:A15)-1}
second string	{=RIGHT(A2:A15,LEN(A2:A15)-SEARCH("+",A2:A15)-1)}

After we have the two strings in separate fields, students should decide on the separator characters and type a space, a hyphen or leave the cells empty in Column D. Conditional formatting helps in differentiating cells with space and empty ones (orange and green, respectively). In conditional formatting, we should formulate a *yes/no* question (a formula) and set up a formatting (actually, a colour) in case the answer is *yes*. *Figure 21* shows two questions and two colours, as cells containing a hyphen are not coloured:

– If the cell contains space, the cell is filled in orange.
– If the cell is empty, it is filled in green.

However, when a space character is found in a cell, there is no need to check whether the cell is empty or not. This option can be set up with the *Stop If True* check box.

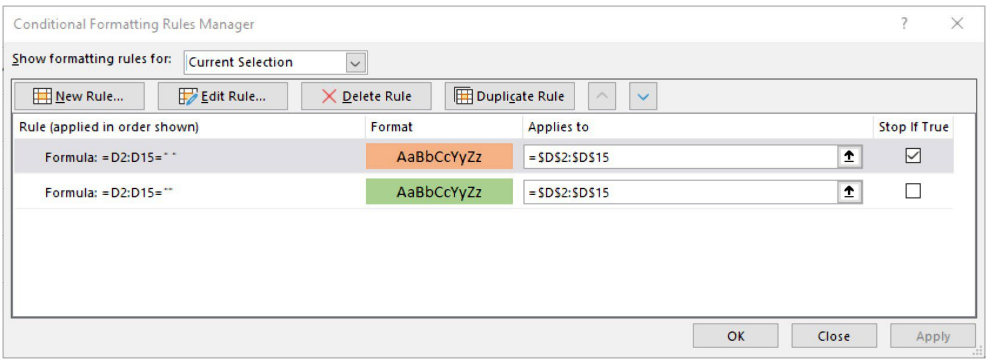

Figure 21. *Two formulas to set two different colours which show the difference between the empty and the space cells*

We are prepared to concatenate the first string, the typed character, and the second string in the Field *solution* (*Table 3*).

Table 3. *The concatenation of the three strings*

solution	{=B2:B15&D2:D15&C2:C15}

However, the field *solution* does not display the right answers. There are cases when a capital letter appears in the middle of the word (*Figure 19* Column E, field *solution* Cells E12 and E15). If we analyse the results, we can reveal that the original task is tricky, and expects the students to debug their results. The same is required in the spreadsheet, where an automated solution is needed to avoid these errors.

The solution is to change the first letter of the second string to lower case whenever the two substrings (field *typing* is empty) make up one word (*Figure 22*):

– in order to change the initial letter of the second string to a lower case or not, we must know whether the cell in field *typing* is empty or not;

– in order to know whether the cell in field *typing* is empty or not, a *yes/no* question must be formulated;

– in order to have lower cased second strings, they should be changed to lower case.

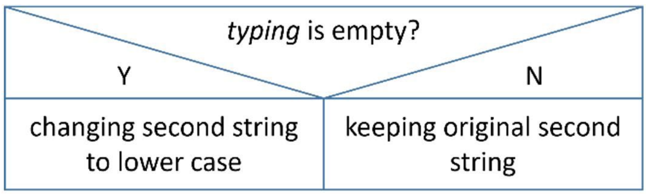

Figure 22. *The algorithm of changing the second string to lower case*

After building the lower-case algorithm, the coding can be carried out (*Table 4*).

Table 4. *Changing the second string to lower case of field typing is empty*

lower case	{=LOWER(C2:C15)}
yes/no question	{=D2:D15=””}
conditional changing to lower case	{=IF(D2:D15=””,LOWER(C2:C15),C2:C15)}

The final step in the digital solution is to concatenate the original first string, the typed character, and the modified second string (*Table 5*).

Table 5. *The concatenation of the three strings with the modified second string*

corrected solution	{=B2:B15&D2:D15&F2:F15}

4. Conclusions

For the present paper, grammar tasks from traditional paper-based and PDF format course books and exercise books were selected, for which digital solutions were offered at various levels. Our goal was to demonstrate how TPCK would serve developing both content and digital knowledge, along with problem-solving abilities. These presented tasks have been carried out during either informatics classes or other, such as language, classes tutored by primary school teachers.

The listed examples and their solutions show that various digital tools can be applied where the original paper-based solution might slow down the thinking process and switch the focus from the content to the tool. Another advantage of the digital solutions is that students can be supplied with real contents in informatics classes. With this solution, the decontextualized and/or meaningless tasks (as sources on informatics often label them) can be eliminated from the class activities, making students more engaged in real-world problems matching their best interest. While some might have dissenting opinions about digitalization in schools – arguing that young people overuse online content, and their mobile phones are only used for the social media –, it is our firm belief that there is a huge difference between meaningless and unrestricted/unlimited time spent on digital content and value-added, productive use of computers. Furthermore, while students may have access to the computers during the entire class, in the presented approach digital tools are only used occasionally, when the tutors require so in the course of teaching.

In the long run, we hope that investing into effective, good-quality, and fast problem solving at an early age will return if these skills are used for life-long digital activities. Although many laypeople think that these tasks are easy to solve, it has been demonstrated many times that this is not the case, and ad-hoc solutions are not effective and efficient.

In our approach, data analysis and repetition (standardization) take a crucial role. As the presented solutions indicate, a thorough analysis of data is required during digital data manipulation. This is only possible with human intelligence, leading to create error-free automated solutions by explaining the "dumb" computer how to carry out the task. On the other side, repetition while analysing the content, setting up the kanban, building the algorithm, completing the coding, and ultimately checking the correctness of the outputs is crucial in the problem-solving process of a pull education system. This way, the content is repeated so many times that it helps to build up knowledge in the long-term memory.

References

Angeli, Charoula–Nicos Valanides. 2015. *Technological Pedagogical Content Knowledge Exploring, Developing, and Assessing TPCK*. New York: Springer US.

Bényei, Tamás Jánosné. 2023. *Digitális lehetőségek az ötödik évfolyamos magyartanításban* [Digital Possibilities in Teaching Hungarian to Fifth Graders]. Unpublished dissertation. University of Debrecen, Debrecen, Hungary. *https://hdl.handle.net/2437/354105* (Last accessed: 9 August 2023).

Bringhurst, Robert. 2004. *The Elements of Typographic Style*. Point Roberts, WA: Hartley and Marks Publishers.

Brydon-Miller, Mary–Davydd Greenwood–Patricia Maguire. 2003. Why action research? *Action Research* 1(1): 9–28.

Csapó, Gábor–Katalin Sebestyén–Mária Csernoch–Kálmán Abari. 2020. Case study: Developing long-term knowledge with Sprego. *Education and Information Technologies* 26(1): 965–982.

Csapó, Gábor–Mária Csernoch–Kálmán Abari. 2019. Sprego: Case study on the effectiveness of teaching spreadsheet management with schema construction. *Education and Information Technologies* 25(3): 1585–1605.

Csernoch, Mária. 2014. *Programozás táblázatkezelő függvényekkel; Sprego: Táblázatkezelés csupán egy tucat függvénnyel* [Programming with Spreadsheet Functions: Sprego]. Budapest: Műszaki Könyvkiadó Kft.

—— 2021. The stepchild of informatics education: File management. *Academia Letters* 1–4. *https://doi.org/10.20935/al2295* (Last accessed: 9 August 2023).

Eilks, Ingo. 2018. Why a new journal in science education? Why focusing on action research? *Action Research and Innovation in Science Education* 1(1): 1–2.

Ékesné, Baranyai Enikő. 2023. *Digitális feladatok magyarórákon az általános iskola 6. évfolyamán* [Digital Tasks for Hungarian Classes in the Sixth Grade]. Unpublished dissertation. University of Debrecen, Debrecen, Hungary. *https://hdl.handle.net/2437/354108* (Last accessed: 9 August 2023).

Fülöp, Mária–Szilágyi, Katalin. 2021. *Nyelvtan és helyesírás 2. munkafüzet* [Grammar and Spelling 2. Exercise Book]. Budapest: Oktatási Hivatal.

—— 2022. *Nyelvtan és helyesírás 3. Tankönyv* [Grammar and Spelling 3. Exercise Book]. Budapest: Oktatási Hivatal.

Ghazi-Saidi, Ladan–Ana Ines Ansaldo. 2017. Second language word learning through repetition and imitation: Functional networks as a function of learning phase and language distance. *Frontiers in Human Neuroscience* 11: 1–13.

Groundwater-Smith, Susan–Anna Campbell. 2010. *Action Research in Education*. Newbury Park, CA: SAGE Publications Ltd.

Gyarmati, Dezső. 2022. *KemIKT az általános iskolában* [KemIKT in Elementary School]. Unpublished dissertation. University of Debrecen, Debrecen, Hungary. *http://hdl.handle.net/2437/330497* (Last accessed: 9 August 2023).

Kahneman, Daniel. 2011. *Thinking, Fast and Slow*. New York: Farrar, Straus and Giroux.

Kamarás, Krisztina. 2022. *Az Excel program felhasználási lehetőségei az általános iskola felső tagozatán tanórai keretek között* [Possibilities of Using the Excel Program in Upper Elementary Grades]. Unpublished dissertation. University of Debrecen, Debrecen, Hungary. *http://hdl.handle.net/2437/330496* (Last accessed: 9 August 2023).

Kormány, József Róbert. 2022. *A GeoGebra felhasználásának néhány lehetősége a geometria tanításában az általános iskola 5. évfolyamán* [Several Possibilities of Using GeoGebra for Teaching Geometry in the Fifth Grade of Elementary School]. Unpublished dissertation. University of Debrecen, Debrecen, Hungary. *http://hdl.handle.net/2437/330645* (Last accessed: 9 August 2023).

Krafcik, John F. 1988. Triumph of the Lean Production System. *Sloan Management Review* 30(1): 41–52.

Kirschner, Paul A.–John Sweller–Richard E. Clark. 2006. Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching. *Educational Psychologist* 41(2): 75–86. *https://doi.org/10.1207/s15326985ep4102_1* (Last accessed: 9 August 2023).

Kirschner, Paul A.–Pedro De Bruyckere. 2017. The myths of the digital native and the multitasker. *Teaching and Teacher Education* 67: 135–142. https://doi.org/10.1016/j.tate.2017.06.001 (Last accessed: 9 August 2023).

Liker, Jeffrey K. 2004. *The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer*. New York: McGraw-Hill.

Maurer, Robert. 2013. *The Spirit of Kaizen: Creating Lasting Excellence One Small Step at a Time*. New York: McGraw-Hill.

—— 2014. *One Small Step Can Change Your Life: The Kaizen Way*. New York: Workman Publishing Company.

Microsoft Support. Guidelines and Examples of Array Formulas. 2023. *Microsoft Support*. *https://support.microsoft.com/en-us/office/guidelines-and-examples-of-array-formulas-7d94a64e-3ff3-4686-9372-ecfd5caa57c7* (Last accessed: 20 May 2023).

Mishra, Punya–Matthew J. Koehler. 2006. Technological pedagogical content knowledge: A framework for teacher knowledge. *Teachers College Record: The Voice of Scholarship in Education* 108(6): 1017–1054. *https://doi.org/10.1177/016146810610800610*. (Last accessed: 9 August 2023).

NAT2020: Magyar Közlöny, A Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról szóló 110/2012. (VI. 4.) Korm. rendelet módosításáról 290–446 (2017). Budapest. *https://magyarkozlony.hu/dokumentumok/3288b6548a740b9c8daf918a399a0bed1985db0f/letoltes*. (Last accessed: 9 August 2023).

Ohno, Taiichi. 1988. *Toyota Production System: Beyond Large-Scale Production*. Milton Park, Abingdon-on-Thames, Oxfordshire, England, UK: Productivity Press.

Panko, R. Raymond. 1998. What we know about spreadsheet errors. *Journal of Organizational and End User Computing* 10(2): 15–21.

—— 2012. Introduction to End User Computing: The Dark Matter (and Dark Energy) of Corporate IT use and user (U2) Studies Minitrack. 45th *Hawaii International Conference on System Sciences*: 1–24.

Polya, George. 1945. *How to Solve It*. Princeton, NJ: Princeton University Press.

Rother, Mike. 2010. *Toyota Kata: Managing People for Improvement, Adaptiveness, and Superior Result*s. New York: McGraw Hill.

Sápiné Bényei, Rita. 2021. *Magyar nyelv munkafüzet 6* [Hungarian Language Exercise Book 6]. Budapest: Oktatási Hivatal.

Sestoft, Peter. 2011. *Spreadsheet Technology*. Copenhagen: IT University of Copenhagen.

Sweller, John–Paul Ayres–Slava Kalyuga. 2011. *Cognitive Load Theory*. New York: Springer.

Szűcs, Edina. 2023. *Digitális oktatás alsó tagozatban* [Digital Education in Primary School]. Unpublished dissertation. University of Debrecen, Debrecen, Hungary. https://hdl.handle.net/2437/354111. (Last accessed: 9 August 2023).

Womack, James P.–Daniel T. Jones. 2003. *Lean Thinking: Banish Waste and Create Wealth in Your Corporation*. New York: Simon & Schuster.