

AN ANALYSIS OF BEBRAS CONTEST TASK CATEGORIES AND THEIR SUPPORT FOR THE INFORMATICS CURRICULUM IN THE CZECH REPUBLIC

Jan BERKI, Technická univerzita v Liberci, Česká republika

Patrik KLOFÁČ, Jihočeská univerzita v Českých Budějovicích, Česká republika

Tomáš PRŮCHA, Západočeská univerzita v Plzni, Česká republika

Přijato: 3. 6. 2025 / Akceptováno: 3. 7. 2026

Typ článku: Teoretická studie

DOI: 10.5507/jtie.2026.005

Abstract: Teachers, mainly those yet to qualify, need to be inspired by high-quality tasks that develop and/or assess computational thinking. In other words, the informatics content of the curriculum. We aim to define categories that would classify contest tasks into suitable thematic units, and at the same time, which would allow teachers to compile their own sets of tasks. A conceptual analysis of various texts was carried out. The results were compared, synthesised, and consequently verified, with a sample set of tasks. A second conceptual analysis of the tasks was rerun to verify the categorisation. The categorisation process is not always straightforward. It is difficult to assign a task to a particular category. There are many tasks that belong to several categories. The number of tasks that focus on user skills is gradually declining. The categories' classification helps identify contest tasks that reflect the thematic units of the Informatics curriculum in the Czech Republic (CR). It further helps teachers recognise which tasks are suitable for each thematic unit.

Key words: task categorisation; Bebras tasks; computer science; informatics; methodological support; the Czech Republic.

1 Introduction

The Minister of Education, Youth, and Sports issued the Framework Educational Programme for Basic Education (FEP BE) at the end of 2004. Consequently, primary and lower secondary school education changed from a one-level to

*Autor pro korespondenci: pklofac@pf.jcu.cz

a two-level curriculum. The stages of pre-school and tertiary education gradually followed suit. In accordance with the FEP BE, schools have to prepare their own specific School Educational Programmes (SEP). Those were applied for the first time in 2007. However, some amendments took place in the following years. One of the most significant amendments was the introduction of a second foreign language in 2013. Due to this amendment, the content of some educational areas was also modified. The second significant amendment of the national curriculum was the promotion of inclusive education in 2016. Neither of these amendments were used to innovate the Informatics curriculum, despite the fact that an ICT panel was working on the innovations at the time.

The educational area of informatics, up to 2020, mainly focused on digital literacy learning (Information and Communication Technologies, 2007)¹.

The thematic units were:

- The basics of working with a computer,
- Information searching and communication,
- Information processing and application.

The need for the change originated from the document “Digital Education Strategy”, in which one of the priorities listed is the promotion of computational thinking among learners (Pavlicova, 2018, p. 5). The document “The creation and revision of pre, basic, and secondary school education curricula at the national level” was approved in 2016. Therefore, the work on the revision and innovation of the informatics curriculum began. A draft² that draws on two fundamental concepts was approved at the beginning of 2021 (Cernochova, 2019, p. 565):

- The educational area of Information and Communication Technologies (focused on literacy) is changing to the educational area of Informatics, whose primary goal is the development of computational thinking
- Digital literacy is defined as a key competence, and will be promoted in all educational areas.

This is entirely new educational content. Almost 60% of primary schools in the Czech Republic do not have a qualified informatics teacher (Zatloukal, 2020, p. 56). Therefore, it is essential to create methodological support to enable even

¹ The newer version is not available in English. However, this part of the publication has not been significantly changed in terms of definition.

² At the time of writing this article, this version was not publicly available.

unqualified teachers to plan and implement quality teaching of informatics topics. For that purpose, a set of teaching materials was created within the “Support for development of computational thinking” project (PRIM) (Cernochova, 2019, Appendix 1).

Sample tasks have proven successful in presenting the new computational content. Supposedly³, with knowledge of the sample tasks, teachers can better comprehend the content, the difficulty, and the depth of the given topic. Based on the discussions with the Czech School Inspectorate representatives, it became evident that sample testing tasks had to be created. Such tasks can be used in the lessons directly, or can serve as a source of inspiration for creating ones’ own tasks. In addition to the teaching materials, one can use the Bebras contest. The Bebras contest was adapted to the Czech environment in 2008 (Tomcsányi & Vanicek, 2009, p. 214), and all usable tasks can be found in the contest archive⁴. Some of the tasks are also embedded in the previously mentioned teaching materials⁵.

The main goal was to define task categories in a way that would allow for linking with the new educational content of computer science, while maintaining the greatest possible compatibility with the existing Bebras contest tasks categorization system (Dagiene & Futschek, 2008). The basic task was then to determine, by classifying the created categories, whether the topics were sufficiently clear, and whether they were represented by a sufficient number of tasks. We consider both of these factors to be important for the successful implementation of the curricular change.

To achieve the main goal, we set ourselves the sub-goal of finding out how the categories have been defined so far, or in what direction the proposed or implemented adjustments have led. To achieve the basic task, it is necessary to determine the method of assigning tasks, or to compare it with existing known methods. By assigning, it is possible to verify both the method of assigning itself, and the definition of categories itself. It will help to answer the question of whether the categories are well defined, comprehensible, and sufficiently represented, so that teachers can get an idea and build the right concept (preview) of the given issue. However, an interesting question is also whether, and how, the composition

³ The situation in the classrooms is similar.

⁴ E. g. <https://www.ibobr.cz/test/archiv>

⁵ E. g. Berki, J. & Drabkova, J. (2020). *Základy informatiky pro 1. stupeň základní školy* [Basics of Informatics for Primary Schools]. Technical University of Liberec. <https://imysleni.cz/ucebnice/zaklady-informatiky-pro-1-stupen-zs>

of tasks in individual grades and age categories changes precisely in the context of discussions about changes in educational content, or the focus of a given educational area in primary school.

2 Theoretical Framework

When categorising the tasks, various perspectives can be applied. Firstly, it is possible to draw on the definition of Informatics as a field of study. As the context is educational, other natural sources are topics defined in the curriculum document. The contest itself also uses categories. Since the aim was to interlink the tasks with the teaching materials' topics, it was also necessary to draw on materials created for the new informatics content within the PRIM project, which was a system project precisely for the development of a new informatics curriculum in the Czech Republic, and also to support its implementation in schools.

The formulation of categories is based on triangulation, in the form of a combination of data sources (Denzin, 1989, p. 236). The first group of resources focuses on defining the field of Informatics. The second group documents the development of categorization in the competition itself, and the third group of sources analyses changes in the curriculum. The following subsection therefore combines the first two types of sources, and the second then uses the third type.

2.1 Areas of Informatics

In professional texts, the terms themselves were searched for, which could correspond to the areas or topics into which the content of the field of Informatics is divided, as well as the definition or description of these areas.

Machanick (2000, p. 145–146), in his Computer Science curriculum proposal, divides the topics into four areas, according to level of the learner when solving a problem with the aid of technologies. He considers “programming [or rather software solution in all its phases] to be a fundamental tool of the discipline” (ibid.:137). He looks at each topic from different points of view, in other words, from different degrees of abstraction. The following groups of concepts can be identified:

- Solving problems – theory, strategies, analyses, solvability, complexity, computability, verification,

- Theory and terminology – logic, formal models, graphs, trees, automata, Turing's machine, ...
- Development – algorithms and data, paradigms, programming languages, software development phases, databases, operating systems, ...

Schubert & Schwill have a different approach to the definition of informatics. Simply said, they understand informatics as being a discipline that deals with the systematic and automatic processing, storage, and transmission of data in terms of hardware & software, fundamentals, and effect (Schubert & Schwill, 2011, p. 2). The authors use various approaches when defining individual topics. Despite the various forms of definition, software design is considered the fundamental concept (ibid, p. 66); modelling is the second (ibid, p. 75). Formal languages and automata then connect these two pillars. Upon closer examination, we find the following topics:

- Software development – problem analysis, decomposition, problem-solving methods, using algorithms, strategy, operating structures, programming languages, verification, optimisation,
- Modelling – iconic and symbolic models, data types, paradigms,
- Objects – data structures, databases, object-oriented programming,
- Information systems – processes, architecture, networks,
- Formal languages and automata – graphs, trees, automata, Petri net.

Calzarossa, Ciancarini, Mich & Scarabottolo (2011, p. 38), when analysing the Italian curriculum of compulsory and elective informatics courses, distinguish different topics, which can be grouped as follows:

- programming, programming languages,
- webpage design, the internet, and navigation tools,
- hardware, computer networks, and architecture,
- working with various software packages,
- fundamental concepts of informatics.

When comparing the German and British informatics curricula, the research group identifies several topics for a questionnaire survey on which respondents from different countries had to comment. Programming is divided into three topic groups. The first group focuses on concepts, languages, and other areas. The second group focuses on the actual programming necessary to solve a problem. The

third group involves activities connected with the project perspective of project programming. Algorithmizations or robotics stand alone. There are also some topics that more or less overlap with digital literacy, whether in the category of applications (editors), privacy and ethics, or databases. Theory, where automata are mentioned as an example, stand in a particular separate category (Dagiene, Jevsikova, Schulte, Sentance & Thota, 2013, p. 66).

A number of definitions of computational thinking have emerged over the years, suggesting that there is no single definition of the term. For the purposes of conceptualizing IM and its inclusion in education, we should not attempt a definitive definition, but rather seek to find similarities and relationships in discussions of IM (Denning, 2017). We understand computational thinking as a problem-solving process, and draw on the definition by ISTE & CSTA (2011):

- Problem – formulation enabling solution with the help of technologies
- Data – organisation and analysis
- Models and simulation
- Algorithmization
- Optimisation
- Generalisation
- Bebras International Contest

There has been an intention to categorise the tasks right from the outset. The following categories were established in 2007 (Dagiene & Futschek, 2008, p. 21):

- Information comprehension – representation, coding, and encryption,
- Algorithmic thinking, including programming aspects,
- Using computer systems – search engines, email, spread sheets, etc. (general),
- Structures, patterns, and arrangements – combinatorics, graphs, etc.,
- Puzzles – logical puzzles, games,
- ICT and Society – social, ethical, cultural, international, legal issues.

As some countries have revised their informatics curricula and emphasised the development of computational thinking, a proposal for the revision of the categories was put forward. This change was two-dimensional. The first dimension drew on informatics concepts, and can be referred to as the theoretical level. The second dimension builds on computational thinking, and can be referred to as the skills level (Dagiene, Sentance & Stupurienė, 2017, p. 35–36). The first dimension

corresponds to the concept in other cited articles and, the thematic division of the curriculum. For the purposes of this article, we will stick to the following:

- Algorithms and programming, including logical reason,
- Data, data structures and representation – graphs, automaton, data mining,
- Computer processes and hardware – scheduling, parallel processing,
- Communications and networking – cryptography, cloud computing,
- Interaction, systems, and society – all other topics.

Algorithms and programming, as a thematic unit, remain together. Yet, thematically logical tasks were added to them. Perhaps, due to the fact that their solution requires an algorithm, which has to be stated. Topics in other categories were reshuffled. The last category can be called “odds and ends” or “miscellanea”. But recent studies also expand this view by emphasising reliable assessment and broader digital competence frameworks linking computational thinking with informatics and media literacy (El-Hamamsy et al., 2022; Vrabec & Furtáková, 2024).

2.2 New curriculum

The turning point, in terms of the Informatics curriculum content, was the year 2013. England is frequently mentioned in this context.

Additionally, changes in secondary education can also be found, for example, in France. Learners encounter four thematic topics within the framework of the new elective subject. Algorithmization focuses on particular algorithms applied in informatics when searching in fields or graphs (Tort & Drot-Delange, 2013, p. 34). The thematic topic “Languages and programming” involves programming in specific languages, and various concepts, such as functions, types of data, and correction of a programme. In contrast to the definition described in the chapter “Areas of Informatics”, computer and network architecture is considered a unique thematic unit here (ibid.: p. 34–35). According to this syllabus, one must understand data digitalisation, its compression and coding, and Boolean operators, in order to understand information (ibid.: p. 34).

Changes in the Polish curriculum appeared a year earlier. In lower secondary education, algorithms and solving problems with the help of computers within computational thinking are the main focus, yet programming itself is not included in the curriculum (Sysło & Kwiatkowska, 2013, p. 45). The educational content in Polish secondary schools builds on seven thematic units (ibid.: p. 47):

- Security
- Processing information
- Communication and co-operation
- Using applications
- Algorithmization and problem-solving
- Digital learning
- The social aspect of ICT

As described in the introduction, Informatics as an educational domain focuses on different areas in the Czech curriculum, and therefore it has a different educational content. Cernochova (2019, p. 565) states that the expected outcome and teaching materials are divided into four thematic units. These units are the same in primary and secondary education, so that one can work on and observe the development continuum.

- **The Data, Information, and Modelling** unit gradually introduces learners to the co-existence of information and data. They explore ways to code information in order to store and share it, they create models of various abstraction levels, and then apply them, while solving various problems.
- Learners learn to read, create, check, correct, finalise, and optimise algorithms and programmes. They identify and use programme constructions in the thematic unit **Algorithms and Programming**.
- **The Information Systems** unit helps learners to recognise particular system features and their relations. Learners put data into tables, and interlink it within databases. They deal with users, their rights and needs, and the consistency and security of data.
- **The Digital Technologies** unit enables learners to understand digital technologies. Based on that knowledge, they comprehend the world around them better, and they grasp the regularities of other thematic units.

Similar units can be found in new teaching materials, created mainly within the PRIM⁶ project, which are based on the draft of the new IT curriculum in the Czech Republic, and which are intended to serve as support for teachers. The basic requirement was to create material that would be comprehensible even to teachers with other qualifications outside of IT.

⁶ Available in the Czech language at <https://imysleni.cz/ucebnice>.

Of particular note is the textbook on robotics, which is an optional part of the curriculum. It develops similar competencies as the programming textbooks, but requires the purchase of robotic toys or building kits. It is therefore up to the school to decide whether, and how, to include this topic, as well as whether, and how, to make the necessary investment. The second special textbook is material focused on larger programming projects, which motivates students to acquire more in-depth skills in problem analysis and demography.

3 Methods used

The task categorisation was conducted in two phases: (1) defining the categories, and then (2) assigning the tasks.

The methodology of defining and determining categories is based on conceptual text analysis, which examines a selected concept and also focuses on examining the occurrence of selected concepts in a texts (Bush et al., 1994–2012). The analysis was executed upon two primary sources – the new national Informatics curriculum and the course books created within the PRIM project. The first phase of the analysis focused on the Key Word In Context occurrences of the lemma “topic”. The second analysed element was the lexical expression “teaching material”, and the subsequent occurrence of keywords or key terms. The key terms identified during the first phase of the analysis served as navigation items. The research primarily focused on the names of the topics, materials, or on particular academic key terms.

The second step of defining categories was focused on “verification”. Again, the searched-for highlighted term was the lemma “topic”. Apart from helping identify other topics which had not been detected in the group of primary sources, the findings aimed to detect frequent clusters of terms in particular topics. This clustering was helpful when describing the defined thematic categories and assigning tasks, namely during determination of the typical features of a category by which the task could be categorised. This group of texts is comprised of academic articles, the main ones of which are listed in the chapter “Theory”. It should be noted that some categories are difficult to assign, depending on the approach to solving the problem, so some flexibility is required (Dagiene, 2017). A broader definition of categories corresponding to the thematic units of the new informatics curriculum in the Czech Republic would actually facilitate this situation. On the other hand, it could lead to a less clear description of individual concepts. The

difference between some would be erased, as the tasks corresponding to them would be included in one common category.

The created categories were labelled and defined by the key terms and the characteristic features of the tasks. Two independent researchers then independently conducted an initial categorization of the tasks from the first year, 2008, for the “Benjamin” and “Junior” age categories, for a total of 34 unique tasks. These two researchers work at university departments that provide informatics teacher education programmes, and also serve as informatics teachers at the lower and upper secondary school levels. The aim was to verify whether the characteristics of the thematic categories were sufficient enough to categorise the task at hand. It was further discussed whether the defined categories were clear, whether to divide or combine some categories, or to create new ones. After this verification phase, each category was labelled with a three-letter code or, in the case of tasks that by their focus fulfilled several categories, the given tasks were assigned to two categories (e.g. ALG/MOD).

When analysing the individual tasks, the wording or the graphical features of the tasks were taken into consideration, as well as the answer explanation of Bebras tasks proposed by their authors, and the fact whether the contestants were explained how the task was linked to informatics. The same two researchers from the initial categorization then individually categorised all of the tasks from all of the age categories between the years 2008–2019. There were a total of 772 individual tasks. The final assigned codes were compared. If the codes of the tasks matched, then the code was permanently assigned. If the code did not match, a third researcher, who specialises in teaching informatics, assigned a code. If such an assigned code matched one of the previously assigned codes, this code was permanently assigned. The permanent code was assigned, based on discussion, assuming it did not match any of the previously assigned codes. After the initial assignment, the researchers completely agreed on 59% of the tasks, in 24% there was a partial agreement (when assigning 2 categories), and they completely disagreed in 17% of all tasks. “The agreement of experts on less than two-thirds of the tasks confirms the difficulty of categorization” (Ternik et al., 2020). Comparable methodological approaches have been also applied to structure computational thinking assessment and validate categorisation consistency (Guggemos & Seufert, 2023).

4 The course and results of problem solving

This section provides an overview of the problem-solving process and the analytical procedure used to evaluate the collected tasks. It introduces the structure of task categories, explains the principles behind their classification. The goal is to clarify how the results were obtained and how they relate to the research objectives.

4.1 Task categories and their definitions

There are two types of research outcomes from this study. The first outcome is the design and the definition of suitable task categories in the context of teaching support, so-called “new informatics”. The second outcome is the demonstration of how many tasks appear in each category, and their involvement within the time and age categories of the contest.

The categories with their codes used during the categorisation process are listed below:

- encoding (KOD)
- encryption (KPT)
- data representation (MAT)
- digital literacy (DIG)
- models (MOD)
- algorithms (ALG)
- programming (PRG)
- information (INF)

As can be seen from the above list, it was decided that the categories would be defined with a more detailed division than corresponds to the thematic units of the new informatics curriculum in the Czech Republic. But, it is linked to the specific expected output of this curriculum. Each of the categories is first defined by a description of the concept. The following are examples of keywords, as well as a sample task.

Encoding – KOD

This is the process of translating data from one form to another. The number of code characters, as well as the probability of their occurrence and frequency in the codeword, may change during the transition from one code to another. Furthermore, a change in redundancy may also appear (Dumas, Roch, Tannier, Varrette, 2015). The coding process can take place because of different reasons, such as the transmission, sharing, or storing of information.

According to FEP informatics, the student proposes and compares different methods of encoding data, with the aim of storing and transmitting it (RVP, 2021). The tasks assigned to this category, in principle, draw on the generally accepted coding algorithm (it is not about securing the data), which is consequently applied when completing the tasks.

The characteristic features of the KOD are: the occurrence of key words, which contain coding letters, gestures, words, pictures, a communication code, etc. Key word conversion into another previously defined form.

The following task shows the typical features of conversion from one form to another. The typical word code, which can be found below the task for clarification, is not used in the instructions.

Digital numbers and diodes, code 2019-MY-02

Alena wants to light up a number with the help of LEDs. Each digit of the number consists of 7 diodes, labelled A, B, C, D, E, F, and G.

The diodes can be operated via a simple programme. The example shows the programme input, and the consequent lightning of the diodes for the number 103:

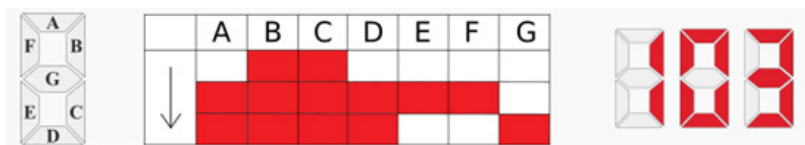


Image 1 Archive – Kadet 2019 – Digital numbers and diodes

How will the diodes light up when Alena inserts the following input? Click on individual digits and set the number.

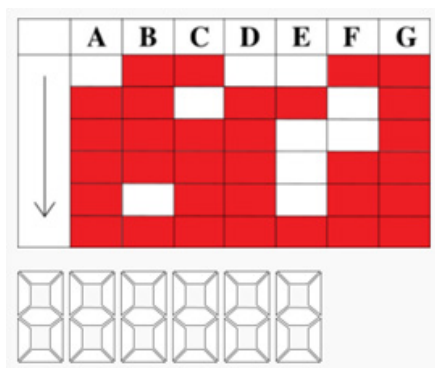


Image 2 Archive – Kadet 2019 – Digital numbers and diodes

Argument: In this task, each digit is encoded using the letters A, B, C, D, E, F, and G, depending on whether the particular diode lights up or not; this code is consequently recorded in the table for clarity. The digit 4, for example, is coded as BCFG and the digit 2 as ABDEG. This coding indicates which diodes should light up (Kadet 2019 – Digital numbers and diodes).

Encryption – KPT

The tasks in this category draw on the secret content of the given information. A plain text is coded to a cipher text, and vice versa (Bauer, 1997).

We do not want our messages to be easily read when they are intercepted, as they may contain passwords and/or other private information. Therefore, the messages are encrypted, which enables them to be turned into secret messages. The recipient of the secret message must be able to decrypt it and reveal the original wording. If an unauthorised person detects such a secret message, they will not be able to decrypt it. According to the FEP of informatics, the student stores and manages his/her data in a suitable format, with regard to its further processing or transmission (RVP, 2021).

The characteristic features of the KPT are: the instructions contain phrases such as “writing secret messages”, “communication via secret messages”, “finding the correct password”, “encryption table”, “encrypted messages”, etc.

The following task includes the characteristic feature “secret message” already in the heading; consequently, the feature can be noted in the instructions.

Secret messages, code 2016-UK-06

Special agents Boris and Berta communicate via secret messages.

Boris wants to send Berta the following secret message:

SEJDISESMARTINEMO6

He writes each feature of the message into the grid of four columns. He writes from left to right and starts from the top. If some empty spaces appear, he fills them in with an X.

Then, he encrypts the message by reading the features by columns from left to right, and top to bottom. He gets an encrypted message:

SIMIOESAN6JEREXDSTMX

Berta then uses the same pattern to answer Boris. The encrypted message she sends is:

RMUAOÍDMZMU!UBTX

What is the answer Berta sent?



S	E	J	D
I	S	E	S
M	A	R	T
I	N	E	M
O	6	X	X

Image 3 Archive – Benjamin 2016 – Secret messages

Argument: Encrypting and decrypting (deciphering) messages is carried out with the help of ciphers. There are many ciphers. The cipher used in this task is the transposition cipher. Its customary function is to reshuffle the positions of

lines and columns (lines for columns and columns for lines). Consequently, the messages are inserted into the grid. The science of ciphers is cryptography, and modern cryptography is a vast research area that includes many other ciphers built on complex mathematical theory/theories (Benjamin 2016 – Secret messages).

Data representation – MAT

The tasks in this category are interlinked with the mathematical presentation of data. Learners work with tables and mathematical graphs. It is crucial to read the information in tables and graphs correctly, and process it when solving such a task. According to the FEP of informatics, the student sets the display, sorting, and filtering of data in the table in order to be able to answer the question (RVP, 2021).

The characteristic features of the MAT include pictures, tables, or mathematical graphs; without them, the solution of the task would be almost impossible.

The following task involves a typical feature – a table, which the solver refers to in the instructions.

Schoolmates' meeting point, code 2012-NL-03

Six former schoolmates live in different parts of a region (see the map).

The roads in the region are winding. The table shows the distances between their homes. The schoolmates want to meet at one of them. They will all drive.

Where should they meet, so that they drive as few kilometres as possible (all together)?

	Žďár	Lhota	Hodínov	Vilice	Vrbov	Pinovice
Žďár	0	15	45	40	25	35
Lhota	15	0	35	30	25	40
Hodínov	45	35	0	10	40	55
Vilice	40	30	10	0	30	50
Vrbov	25	25	40	30	0	15
Pinovice	35	40	55	50	15	0

Image 4 Archive – Kadet 2012 – Schoolmates' meeting point

Argument: The information presented in the table enables a clever calculation of the overall shortest possible distance. The learners should come up with the simple solution of adding the numbers in a line or a column. Storing data and correctly reading it is one of the main topics in informatics (Kadet 2012 – Schoolmates’ meeting point).

Digital literacy – DIG

Digital literacy is defined as a set of digital competencies (theory, skills, attitudes, and values), which an individual needs in order to safely, confidently, critically, and creatively apply digital technologies in various environments such as work or leisure, when learning, at leisure, and even during social participation (Ferrari, 2013). Information literacy focuses on working with information with the appropriate use of digital technologies. Digital literacy is a part of this.

We have assigned the DIG code to Bebras contest tasks that involve working with computers (HW, SW), appropriate behaviour on the Internet, differentiating the types of files, and user procedures. According to FEP informatics, the student uses functions for automating data processing (RVP, 2021).

The characteristic features of the DIG are: tasks with facts about HW and SW. Questions such as “How large is the RAM/ HDD of a computer?” “Is it legal to...?” “What type of applications ...?” “What will happen if...?”

The following task consists of a typical feature. The instructions include the lexical expression “text editor tool Find”, it is SW and its tool “Find and Replace”.

Search and replace 2, code 2013-FR-10

Beaver has a file that contains many lines, all having the same format, as shown in the left column below. He would like to transform these into the format of the right column.

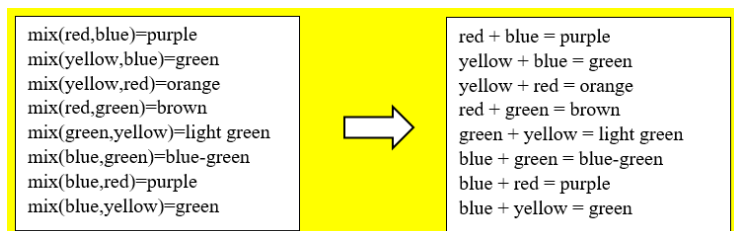


Image 5 Archive – Senior 2013 – Search and replace 2

To help him, you have a basic search and replace tool. If you give a sequence of characters to find, for instance '5)', and a replacement sequence of characters, for instance '9)' and click on "Replace all", the tool will replace every occurrence of the first sequence with the second one.

Change the text into the required form.

- The button [Replace all] may be used multiple times. You can replace the text in several steps.
- The computer automatically checks whether the task has been completed. After pressing the button [Again], the original text appears.

Argument: The function "Find and replace" can save a lot of work – the longer the text, the greater the time efficiency. A skilled user thinks first what they want to find and what they want to replace it with. An inefficient user will start changing the text manually (Senior 2013 – Search and replace 2).

Models – MOD

In this category, there are tasks that touch on informatics models. Informatics have a variety of modelling methods available, including logical structures, graphs, automata models, Petri nets, UML diagrams, etc. Frey (1983) gave an approximate classification of models (a more comprehensive classification is given by (Thomas, 2002)), distinguishing between iconic models that refer pictorially to (the original) and symbolic models that are described using a formal language with well-defined syntax and semantics.

- Iconic models serve mainly to illustrate facts, but usually do not explain them, due to the absence of regularities or connections.
- Symbolic models represent the general form of models. They are more difficult to imagine, but provide an adequate explanation rather than a description.

The goal of information modelling is usually the solution of a problem using iconic symbols (trees, graphs, structograms) and symbolic models (programmes, grammar) as intermediate steps on the way to the final result (Schubert & Schwill, 2011).

A model is a collection of subjects, concepts, or thoughts gathered in a form with relations different from the modelled object. A model represents a given investigated object. Models are usually created to understand the investigated

object; therefore, it is important to use simplification, due to the object complexity (Robert E. Shannon, 1998). Models are used for the presentation of reality, while omitting irrelevant information. Tasks focusing on analysing graphs, optimising, and visualising relations between objects fall into this category. According to the FEP of informatics, the student evaluates whether the model contains all of the data necessary to solve the problem. He/she defines the problem and determines what information he/she will need to solve it (RVP, 2021).

The characteristic features of the MOD are: words, graphs, and/or models referring to pictures with various nodes and paths, appear in the instructions. The tasks frequently contain optimisation of searching for variants, which are: the shortest, the cheapest, the largest, the longest, the number, etc.

The following task does not include the word graph or model. Nevertheless, the instructions encourage orientation in a picture. The typical feature significant for the picture is the directed graph.

Toll roads, code 2018-IR-06

Bob decided to drive from Hamper to Mug. In the map below, circles with letters are cities or towns, and lines are two-way roads. Roads have junctions in places of intersections. The numbers beside any road is the toll that cars must pay every time they use the road. Cars can change the path at junctions, but they need to pay the full toll for the road they enter. For example, to drive from city B to city C you can take road 18 and road 6, and therefore pay $24 = 18 + 6$ toll.

How much is the minimum in tolls that should Bob pay to drive from Hamper to Mug?

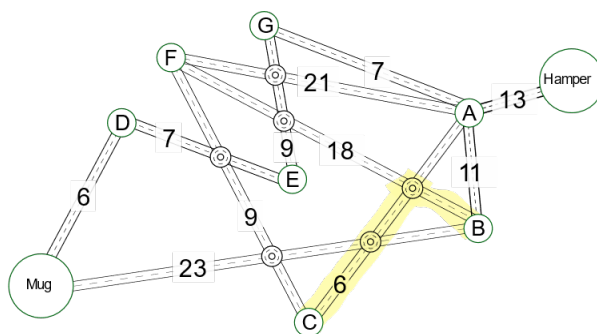


Image 6 Archive – Junior 2018 – Toll roads

Argument: In graph theory, the shortest path problem is the problem of finding a path between two vertices (or nodes) in a graph so that the sum of the weights of its constituent edges is minimized. The graph's vertices correspond to inter-sections, and the edges correspond to road segments, each weighted by either its length, its toll, its traffic load, or its time, etc. (Junior 2018 – Toll roads).

Algorithms – ALG

An algorithm can be defined as a strategy that establishes a clear step-by-step procedure for solving a given problem. In fact, we could argue that it is planning (Jawaharlal, 2016). Although today, this term is mainly used in informatics and natural sciences in general, its scope is much broader (food recipes, instructions, and procedures...) (Cormen, 2009).

In our case, the tasks from the Algorithms category do not require knowledge to write the code itself, in the sense of programming language syntax. Code here means the actual procedure. The tasks also contain “programmes” in the natural language of the pupils. Also, according to the FEP of informatics, the pupil divides the problem into individually solvable parts and proposes and describes the steps to be taken to solve them (RVP, 2021). In this way, the category and the assignment of tasks were viewed.

The characteristic features of the ALG are: finding a number, a sequence, the fastest, the shortest, the smallest, the largest, ...variants of the possible result.

The following task includes the characteristic feature, finding the smallest possible number of jumps.

Froggie, code 2013-SE-05

The frog jumps from one water lily leaf to another. It can jump a long jump (five leaves maximum) or a short jump (two leaves).



Image 7 Archive – Benjamin 2015 – Froggie

It falls into the pond at places where the leaf is missing.



Image 8 Archive – Benjamin 2015 – Froggie

What is the smallest possible number of jumps to reach the last water lily leaf in the picture?

Argument: Find the smallest number of steps to proceed from one place to another (for example, which route to walk from one house in the city to another, or the fastest possible route when delivering mail). Algorithms to solve complex problems efficiently have been invented. Computers can then solve such problems in a very short time. One such algorithm is a width search (Benjamin 2015 – Froggie).

Programming – PRG

Programming is an informatics process starting from suggestions of how to solve a problem with the help of computing technology, and ending with a usable computing programme. It involves analysing the problem, comprehending it, finding a suitable algorithm, and writing the source code in the programming language (Bebbington, 2014).

Within our categorisation, this concept is perceived in a narrower sense. The tasks in this category contain an algorithm code that is the key element of the tasks and their solutions. Furthermore, the tasks require the skills necessary to assign variables and use various symbols correctly. According to the FEP of informatics, the student uses repetition, verifies the correctness of the procedure, and finds and corrects any errors in it (RVP, 2021).

The characteristic features of the PRG are: writing a programme (designing the code), assigning variables, assigning commands, orientating in the commands, and adding various parameters into the programme.

The typical features in the following task represent complete programmes for a robot, which lead to the understanding the operation of the robot, and secondly the completion of the task “writing a programme”.

A drawing robot, code 2013-FR-05a

A drawing robot moves on tiles and paints each field it enters. The following programme operates it:

Program Robot:

4S – the robot moves 4 fields South from the place it is standing on

3E1W – the robot moves 3 fields East and then 1 field West

2S3EN – the robot moves 2 fields South, then 3 fields East and then 1 field North

4(3SE4N) – the robot runs the command in the brackets four times

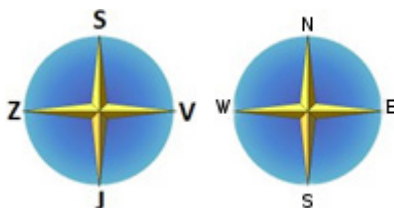


Image 9 Archive – Senior 2015 – A drawing robot

Write a programme for the robot so that it can paint a grey ornament.

Write the programme on the line; a picture will appear after pressing the “Save” button. You can try more than once.

The robot is in a green field at the beginning. If the fields turn green, the robot has painted them correctly; if they turn red, the painting is incorrect. The maximum length of the programme is a combination of ten letters and numbers. You can use spaces, but they are added to the number of tokens. It is not permitted to use brackets when writing the program (()).

Argument: It is necessary to write a programme in a programming language. The successful solver must know how to compile a sequence of commands, and to use their repetition. When writing a programme, it is essential to “save sources”; in other words, to find a programme that will conduct the activity the fastest, with the lowest costs. (Senior 2015 – A drawing robot).

Information – INF

The tasks in this category rely on logic or logical thinking. They involve drawing conclusions from different perspectives, and exploring multiple possibilities based on the task data. The task data is variously visualised. It is presented in

different images. Anyone wants to succeed in the tasks must be able to choose from the offered information, understand the meaning and purpose of this information, and correctly place it in relation to the situation where it should be used. According to the FEP of informatics, the student obtains information from the data, interprets the data, and detects errors in other people's interpretations of the data (RVP, 2021).

The characteristic features of the INF tasks include the following questions "Can you recognise ...?" "Which of these answers ...?" "What has to be done to ...?" "Which entry corresponds with ...?" "Which information is ...?" "Can you recognise which from the following ...?" "Where ...?"

Drawing conclusions from the supplied information and the question "What must be done to....?" are the typical features of these tasks.

What is inside the egg?, code 2012-IT-13

Lenka is looking at four chocolate eggs displayed in a shop window. Two of them are open. Therefore, one can see if there is a surprise in them; however, it is impossible to recognise the cover icing from her perspective. Two eggs are sealed; that is why the colour of the cover icing is visible; however, it is impossible to recognise if there is a surprise inside them.

Lenka assumes that there is a surprise in all the blue eggs.

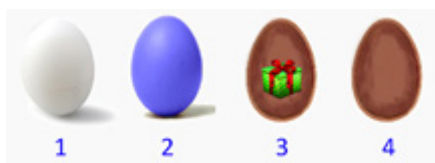


Image 10 Archive – Junior 2012 – What is inside the egg?

What does Lenka have to do to prove her statement?

Argument: Logic plays a significant role in proving the accuracy of the statement made. The task offers different procedures which can lead to proving the given statement. (Junior 2012 – What is inside the egg?).

4.2 Number of tasks in categories

Seven hundred and seventy-two tasks were categorised into five different age group categories. A stack bar graph was used to depict all of the categories which appeared between 2008 and 2018. Assigning the tasks to the categories was a time consuming process which was, to some extent, influenced by the subjective opinion of the authors; that is why a double-check of the assigned category was run. This issue will be dealt with in the chapter “Discussion”. Since some tasks could be assigned to two categories at the same time (e.g. the MOD category and the ALG category), we decided to count these tasks in both categories, however, with only a half mark each. This calculation ensures that the total number of tasks per year corresponds with the number of assigned categories.

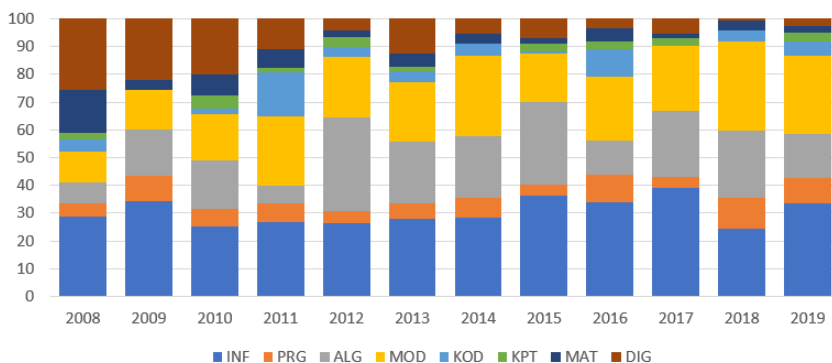


Image 11 Task categories 2008–2019

A linear regression model was developed to determine the year-on-year trend in the individual task categories (Seber & Lee, 2012). The MOD category has a model with the probability value $R^2 = 0.58$. Based on the results of the analysis, the occurrence of each type of task, on average, increases annually by 1.32 percentage points (95% estimation: from 0.54 to 2.11). The DIG category shows a decreasing tendency. With the probability value $R^2 = 0.78$, the developed model confirmed a year-to-year decrease of 2.04 percentage points (95% estimation: from -2.81 to -1.28). The DIG category tasks scarcely appeared (0.7%) in 2018; they slightly returned (2.5%) in 2019. For the other categories, it was impossible to determine

an increasing or decreasing trend with a 95% or 90% estimation when using the linear regressive model.

Category	Absolute numbers	Percentage
INF	234.5	30.4%
MOD	174.5	22.6%
ALG	155.5	20.1%
DIG	69.5	9%
PRG	52.5	6.8%
KOD	34.5	4.5%
MAT	34.5	4.5%
KPT	16.5	2.1%

Tab. 1 Absolute number and percentage of tasks in categories 2008–2019

Table 1 displays the absolute and percentages values of the tasks' occurrences in categories. This is an overall summary of the years 2008–2019.

The category INF is represented in the contest consistently throughout the years. Its share in tasks is continuously high, and overall, it is the most frequently used task category. The ALG category shows some fluctuation in its use (the year 2011 – 6.7%, and the year 2012 – 33.6%), yet, its representation in the contest is stable, and it is the third most numerous category. There is a relatively low representation of the KOD, KPT, and MAT categories. They scarcely appear in some years; for example, the category MAT was only significantly used in the first year of the contest (15.6%), but was not used in the following years.

A relatively low representation can also be observed in the PRG category, which maintains a constant trend across the years. Still, we consider its separation from the ALG category to be important (in total, 52.5 tasks).

5.1 Bebras task age categories

The contest was divided into three age categories in 2008. The original three categories were: Benjamin, Junior, and Senior. In 2010, the category Kadet was added, and in 2012 the category Mini. These five age categories have been used since 2012, without any changes. The age categories of the contest correspond to

the classification of education categories, according to ISCED 2011 (UNESCO, 2012); see the table below.

Bebras task age categories	ISCED classification
Mini	ISCED 100
Benjamin	ISCED 244
Kadet	ISCED 244
Junior	ISCED 344, 354 and 353
Senior	ISCED 344, 354 and 353

Tab. 2 Bebras task age categories connected to ISCED classification

In the ISCED 100 category, which corresponds with the age category Mini, the INF category is used most frequently, followed by the ALG and MOD categories. These categories tend to be used constantly. Other categories, apart from the DIG category, have been used marginally throughout the years. Notably, some categories were not used in some years at all. This is the case of the PRG category, which did not appear in the years 2015, 2016, and 2017. The category KOD did not appear in 2012–2015 and 2017–2018. The DIG category only appeared in 2014 and 2018. The category KPT, within the ISCED 100 framework, does not appear in the contest at all.

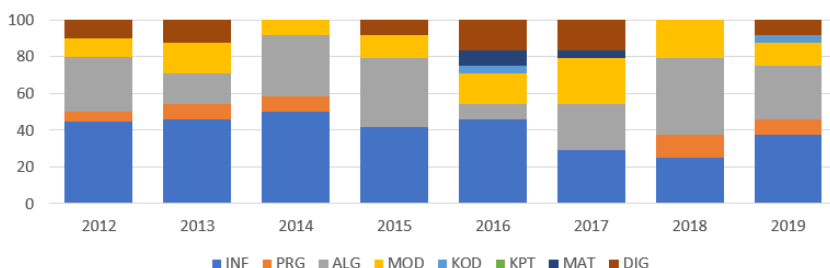


Image 12 Task categories ISCED 100 2012–2019

The category ISCED 244, which corresponds with Benjamin and Kadet's age categories, shows an initial increasing tendency in use within the MOD category, which had a stable representation in the contest during the later years.

A decreasing tendency can be detected mainly in the DIG category. Some categories do not even appear in this ISCED classification. Categories such as PRG (2009, 2010, and 2019), KOD (2009, 2013, 2015, and 2017), KPT (2009, 2014, 2015, and 2018), and DIG (2016 and 2018).

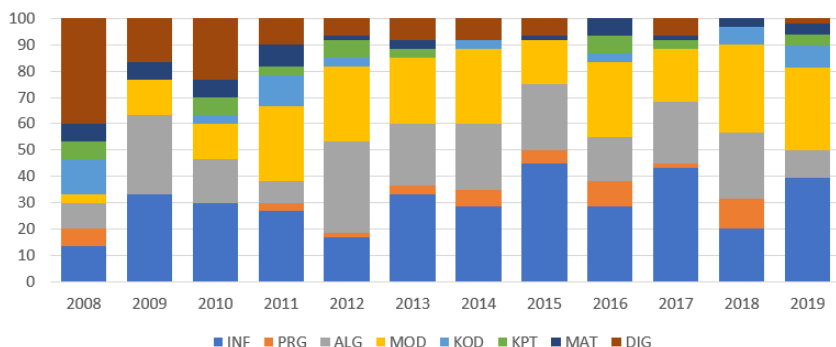


Image 13 Task categories ISCED 244 2008–2019

The categories ISCED 344, 354, and 353, which correspond with the age categories Junior (which partially overlapped with the category ISCED 244 at the beginning of the contest) and Senior, most frequently, recently have made use of the categories MOD and INF. The MOD category shows an initial increasing tendency. There is a visible fluctuation in the use of the ALG category, whose usage grew significantly in the years 2012–2017. However, this cannot be said about 2014; the tasks from the MOD category took over. Some categories do not even appear in this ISCED classification. These are the categories KOD (2008, 2009, 2010, and 2017), KPT (2008, 2009, 2011, 2013, 2014, 2016, and 2018), and DIG (2012, 2017, and 2019).

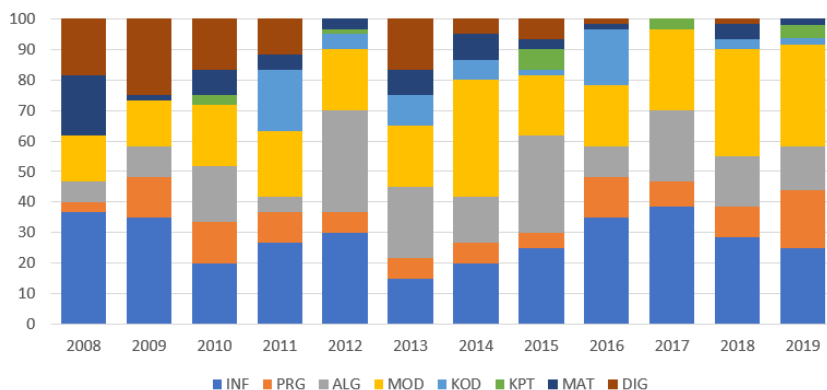


Image 14 Task categories ISCED 344, 355 and 353 2008–2019

5 Discussion

The categories we have defined are aimed primarily at the educational content of the new IT curriculum in the Czech Republic. The Czech curriculum at the national level includes teaching materials. This curriculum is recommendatory, and its purpose is to help teachers connect the expected outcomes, or rather competencies, with specific professional concepts. For this reason, the categories we have defined are conceived thematically rather than activity-wise.

During the categorisation process, several borderline cases emerged, especially between the Models (MOD) and Algorithms (ALG) categories, as well as between Information (INF) and Digital literacy (DIG). Recent assessment studies for example similarly confirmed that modelling and data-representation tasks tend to develop together, as both support abstraction and problem-solving within computational contexts (Guggemos & Seufert, 2023; El-Hamamsy et al., 2022).

To clarify these boundaries, we refined our operational definitions.

- Models (MOD) involve creating or interpreting representations or abstractions of a system e.g. graphs, diagrams, or symbolic models that depict relationships or structures. The focus is on understanding how elements relate, not on the steps taken.
- Algorithms (ALG), by contrast, focus on procedural problem solving a defined, ordered sequence of actions to reach a result or optimise a process.

- Information (INF) tasks emphasise reasoning from or about data interpreting, verifying, or deducing meaning from given information.
- Digital literacy (DIG) tasks target practical use of digital tools and systems, such as searching, communication, or file management.

To ensure consistency in classification, we applied a dominance rule: when a task contained features of multiple categories, it was assigned according to its dominant cognitive objective. A secondary category was only assigned if it was equally essential for solving the task. This procedure helped reduce ambiguity in borderline cases.

Our categories are also more specific than, for example, in the work of Kalas and Tomcsányiová (2009), so there are a larger number of categories. More existing specific categories should better help teachers map them to specific thematic units, or directly to the outputs of the new IT curriculum. In contrast to Dagiene et al. (2017), we decided on a one-dimensional categorization, because we consider it important to help teachers navigate the new content. We therefore primarily wanted to support the understanding of new terms or concepts. When, especially inexperienced teachers, gain more confidence, we can proceed to the second dimension based on operations.

The increasing tendency to use the MOD category was surprising for us, as we expected growth in the ALG and PRG categories. Our assumptions were based on the fact that the emphasis on this issue in informatics has been growing recently, which can be proven by the revision of the informatics curriculum in the Czech Republic. The increasing representation of the MOD category was unexpected. Perhaps the reason for this lies in the presentation of the MOD category tasks, which often rely on visual support (a picture), which is very popular in the contest. For example, in 2019, forty-eight tasks out of seventy-two included a picture in the instructions. What is also important to mention is the content division of the MOD and MAT categories. This was mainly driven by the fact that the content of the MAT category was historically part of the original ICT area in the Czech curriculum, mainly in connection with data processing in a spreadsheet editor. However, it no longer contained the topics that we included in the MOD category, namely finite automata or computer models. This division should therefore work better from the perspective of implementing the new computer science curriculum in the Czech Republic.

It is also essential for us to keep the categories ALG and PRG divided, though we were contemplating using only one category for these tasks at the beginning of our categorisation. We aim to show the teachers who will possibly use the categorised tasks the methodological steps essential for the given categories or areas of informatics. It might not be only the code itself and its formulation. It might be quite important for the teachers to develop their tasks in the ALG category, where no programmes will be necessary. The main aim is to introduce best practice tasks for a given area, though it might seem that the division of the two types of tasks does not play an important role in categorisation.

The year 2018 was also interesting; the category DIG tasks scarcely appeared in that year. Nevertheless, we started to use it again in 2019. A notable fact is that the DIG category does not frequently appear in the ISCED classification 100, except for 2016. The reason for this might be a consequence of different teaching objectives and the contest itself. While the aim of teaching informatics in primary education can be perceived more in terms of acquiring skills in using digital technologies, the contest focuses more on the verification and recognition of principles, which is not easy at this age, and within this thematic category. The contest does not aim to support definition and procedure memorization.

The KOD and KPT categories may be merged with other categories, due to their low statistical significance, which we did not expect when designing them. We originally considered separate categories for this topic to be fundamental. These categories may be merged into one. We kept the two categories, since we expect both topics to be new in informatics teaching, and not entirely clearly defined in terms of the content. Thanks to the examples, though not high in number, teachers can become acquainted with the topics. We may emphasise the difference between the two topics.

The process of assigning the Bebras contest tasks into categories was relatively time consuming, as well as demanding. There were 772 individual tasks. In the beginning, three assessors identified individual categories, which they considered important, and then defined the criteria by which it would be possible to assign the given task to an appropriate category. It turned out that the process was not straightforward, as all the topics are, in principle, interlinked. It is impossible to create a model without coding the information. Likewise, writing an algorithm without a code (regarding the alternative label for programming such as coding) is impossible. When looking for a route in a graph, or solutions in most tasks, we must find the correct procedure. In this sense, a teacher can use one task to

explore the secrets of multiple topics. Our categorisation aims to point out the topic which we consider to be the primary one.

5.1 Limitations

This study has several limitations. First, the categorisation of tasks was partially based on expert judgement, which may have influenced the final classification despite the multi-stage validation process. Second, the analysed dataset consisted only of tasks from the Czech implementation of the Bebras contest between 2008 and 2019, which limits the generalisability of the findings. Third, some tasks naturally span multiple categories, reflecting the interconnected nature of informatics concepts and making strict classification difficult.

From a broader perspective, the proposed categorisation may also have implications beyond the Czech educational context. Although this study focuses on the Czech Republic, the proposed categorisation model can be adapted to other national contexts. The core dimensions – algorithms, modelling, information, and digital literacy are crosscurricular and internationally recognised in frameworks such as the UK Computing Curriculum (2022) or the US CSTA (2023) Standards.

In countries where informatics curricula are structured similarly (e.g., the United Kingdom, Poland, or Slovakia), this model may serve as a reference framework for mapping contest tasks to expected learning outcomes. It may also support comparative international studies examining how national curricula influence task design and the development of computational thinking.

The interconnectedness between the contest, thematic categories, and information system included in the FEP can be perceived as an open question for discussion. Other topics, such as coding, modelling, and algorithms are interconnected. The preconceptions that would lead to the comprehension of the information system can be seen in the suggested category MAT, where the data is structured into tables. The same applies to the INF category, which deals with data presentation and understanding.

From a practical perspective, the categorisation model can support teacher professional development. Each category provides a basis for creating training modules, workshops, or online repositories of Bebras tasks aligned with curriculum outcomes. Integrating the categorisation into teacher education could strengthen educators' ability to recognise computational thinking components and design balanced classroom task sets. This approach is consistent with findings

by Bryndová, Bártek and Klement (2024), who demonstrated that alternative mathematics instruction can foster computational thinking development in primary education.

6 Conclusion

We were able to analyse the historical development of the contest through the structure of task thematic specifications and the trends within this structure, despite the problems mentioned in Methodology and Discussion. The difficulty of categorization is demonstrated by other similarly designed studies that tried to categorize informatics topics. The authors Ternik et al. (2020) talk about a relatively weak agreement between experts when classifying the Beavers' tasks into a category, and who tried to categorize the components of computational thinking. As a result, they did not arrive at a consensual and generally accepted categorization scheme. The Czech Bebras contest tasks can serve as an abundant source of tasks, focusing on models, algorithms, and information processing. Teachers can find examples of tasks in the archive, and other thematic categories that reflect the FEP thematic units. The only exception is the thematic unit "Information systems".

The linear regression model has proven that the number of the category DIG tasks decreases, while the MOD category tasks increase. Some categories experienced fluctuation over some years. The ISCED 100 age category appears to be the least diverse regarding the representation of thematic categories, although we also record mild fluctuations in the internal structure.

Undoubtedly, it could be interesting to verify the applicability of this proposed categorization for the needs of other national curricula. At the same time, it would be interesting to find out whether the composition of the tasks differs in relation to the national curriculum, although an international workshop, whose output is a database of prepared tasks, precedes the actual contest. The categories we have created are sufficiently defined, as evidenced by the consensus of the experts in categorizing the tasks.

Bibliography

- Bauer, F. L. (1997). *Decrypted secrets: methods and maxims of cryptology*. New York: Springer, ISBN 978-3540604181.
- Bebbington, S. (2014). *What is programming* [online]. Available at: <https://yearofcodes.tumblr.com/what-is-programming>

- Bush, C., De Maret, P. S., Flynn, T., Kellum, R., Le, S., Meyers, B., Saunders, M., White, R., & Palmquist, M. (1994–2012). *Writing@CSU: Content Analysis – Conceptual Analysis*. Colorado State University. Available at: <https://writing.colostate.edu/guides/page.cfm?pageid=1309&guideid=61>
- Bryndová, L., Bártek, K., & Klement, M. (2024). *Implikace vlivu alternativních metod výuky matematiky na rozvoj informatického myšlení na základních školách*. *Media4U Magazine*, 21(3), 7–14. ISSN 1214-9187
- Calzarossa, M. C., Ciancarini, P., Mich, L., & Scarabottolo, N. (2011). Informatics education in Italian high schools. In Kalas, I., & Mittermeir, R. T. (Eds.), *Informatics in Schools: Contributing to 21st Century Education*, Berlin: Springer, pp. 31–42.
- Cernochova, M. (2019). Implementing computational thinking into the curriculum and teacher education in the Czech Republic: Facts and first experiences. In *EdMedia + Innovate Learning in Education*, pp. 564–569. Available at: <https://www.learntechlib.org/primary/p/210279/>
- Computer Science Teachers Association. (2023). *CSTA K–12 Standards Revision Overview*. Available at: <https://csteachers.org/k12standards/revision/>
- Cormen, T. H. (2009). *Introduction to algorithms* (3rd ed.). Cambridge: MIT Press, ISBN 9780262033848.
- Dagiene, V., & Futschek, G. (2008). Bebras international contest on informatics and computer literacy: Criteria for good tasks. In Mittermeir, R. T., & Syslo, M. M. (Eds.), *Informatics Education – Supporting Computational Thinking*, Berlin: Springer, pp. 19–30.
- Dagiene, V., Jevsikova, T., Schulte, C., Sentance, S., & Thota, N. (2013). A comparison of current trends within computer science teaching in school in Germany and the UK. *Commentarii informaticae didacticae*, 6, pp. 63–75.
- Dagiene, V., Sentance, S., & Stupurienė, G. (2017). Developing a two-dimensional categorization system for educational tasks in informatics. *Informatica*, 28, pp. 23–44.
- Denning, P. J. (2017). Remaining trouble spots with computational thinking. *Communications of the ACM*, 60(6), pp. 33–39.
- Denzin, N. K. (1989). *The research act: A theoretical introduction to sociological methods*. Englewood Cliffs: Prentice Hall. ISBN 0137743815.
- Dumas, J., Roch, J., Tannier, E., & Varrette, S. (2015). *Foundations of coding*. Hoboken: Wiley. ISBN 9781118881446. Available at: <https://doi.org/10.1002/9781119005940>
- El-Hamamsy, L., Zapata-Cáceres, M., Martín Barroso, E., Mondada, F., Dehler Zufferey, J., & Bruno, B. (2022). *The Competent Computational Thinking Test: Development and validation of an unplugged computational thinking test for upper primary school*. *Journal of Educational Computing Research*, 60(7), 1818–1866. <https://doi.org/10.1177/07356331221081753>
- Ferrari, A. (2013). *DIGCOMP: A framework for developing and understanding digital competence in Europe*. Luxembourg: Publications Office of the European Union. ISBN 978-92-79-31465-0
- Frey, K. (1983). Die sieben Komponenten der Projektmethode – mit Beispielen aus dem Schulfach Informatik. *LOG IN*, 3, pp. 16–20.
- Information and Communication Technologies. (2007). In Jerabek, J., & Tupy, J. (Eds.), *Framework Educational Programme for Basic Education*, Prague: VUR, pp. 32–35. Available at: <https://www.msmt.cz/vzdelavani/zakladni-vzdelavani/framework-education-programme-for-basic-education>
- Guggemos, J., & Seufert, S. (2023). *Computational thinking assessment – Towards more vivid interpretations*. *Technology, Knowledge and Learning*, 28, 539–568. <https://doi.org/10.1007/s10758-021-09587-2>
- ISTE & CSTA. (2011). *Operational definition of computational thinking for K-12 education*. International Society for Technology in Education & Computer Science Teachers Association.

- Available at: <https://cdn.iste.org/www-root/ct-documents/computational-thinking-operational-definition-flyer.pdf>
- Kalas, I., & Tomcsányiová, M. (2009). Students' attitude to programming in modern informatics. *Informática na Educação: teoria & prática*, 12(1), pp. 127–135.
- Machanick, P. (2000). Curriculum 2020. In Greening, T. (Ed.), *Computer Science Education in the 21st Century*, Berlin: Springer, pp. 137–148.
- Pavlicova, V. (2018). *Czech Republic: Country report on ICT education*. Brussels: European Schoolnet. Available at: https://www.eun.org/documents/411753/839549/Country_report_Czech_Republic_2018.pdf
- Rámcový vzdělávací program (RVP) informatika. (2021). *Framework educational program – Informatics*. Prague: Ministry of Education. Available at: <https://revize.edu.cz/files/informatika-2021.pdf>
- Schubert, S., & Schwill, A. (2011). *Didaktik der Informatik* (2nd ed.). Heidelberg: Spektrum Akademischer Verlag.
- Seber, G. A. F., & Lee, A. J. (2012). *Linear regression analysis* (2nd ed.). Hoboken: Wiley.
- Shannon, R. E. (1998). *Introduction to the art and science of simulation* [online]. Available at: <http://www.informs-sim.org/wsc98papers/001.PDF>
- Syslo, M. M., & Kwiatkowska, A. B. (2013). Informatics for all high school students: A computational thinking approach. In Diethelm, I., & Mittermeir, R. T. (Eds.), *Informatics in Schools: Sustainable Informatics Education for Pupils of all Ages*, Berlin: Springer, pp. 31–42.
- Ternik, Ž., Todorovski, L., & Nančovska Šerbec, I. (2020). Assessing the agreement in the Bebras task categorisation. In *Informatics in Schools: Situation, Evolution, and Perspectives*, Cham: Springer, pp. 30–41.
- Thomas, M. (2002). *Informatische Modellbildung – Modellieren von Modellen als ein zentrales Element der Informatik für den allgemeinbildenden Schulunterricht* [Dissertation]. Potsdam: Universität Potsdam. Available at: <http://ddi.cs.uni-potsdam.de/ab/pu>
- Tomcsányi, P., & Vanicek, J. (2009). Implementation of informatics contest Bebras in Czechia and Slovakia. In Mechlova, E., & Valchar, A. (Eds.), *Information and Communication Technology in Education. Proceedings*, Ostrava: University of Ostrava, pp. 214–218.
- Tort, F., & Drot-Delange, B. (2013). Informatics in the French secondary curricula: Recent moves and perspectives. In Diethelm, I., & Mittermeir, R. T. (Eds.), *Informatics in Schools: Sustainable Informatics Education for Pupils of all Ages*, Berlin: Springer, pp. 31–42.
- UNESCO Institute for Statistics. (2012). *International standard classification of education: ISCED 2011*. Montreal: UNESCO Institute for Statistics.
- UK Computing Curriculum. (2022). *Research review series: computing*. London: Department for Education. Available at: <https://dera.ioe.ac.uk/id/eprint/39245/1/computing%20-%20GOV.pdf>
- Vrabec, N., & Furtáková, L. (2024). *Ways of defining digital competences and their components in the EU, EC and UNESCO recommendations*. *Studia de Cultura*, 16(2), 5–17. <https://doi.org/10.24917/20837275.16.2.1>
- Zatloukal, T., et al. (2020). *Kvalita a efektivita vzdělávací soustavy ve školním roce 2019/2020: výroční zpráva*. Prague: Czech School Inspectorate. Available at: [https://www.csicr.cz/cz/Dokumenty/Vyrocní-zpravy/Kvalita-a-efektivita-vzdelavani-a-vzdelavaci-s-\(3\)](https://www.csicr.cz/cz/Dokumenty/Vyrocní-zpravy/Kvalita-a-efektivita-vzdelavani-a-vzdelavaci-s-(3))

⁷ The referenced publication of FEP (RVP) informatics is so far only available in the Czech language (no translation into another language has yet been created).