General Key Concepts in Informatics: Data

Péter Szlávi and László Zsakó

Abstract. "The system of key concepts contains the most important key concepts related to the development tasks of knowledge areas and their vertical hierarchy as well as the links of basic key concepts of different knowledge areas. When you try to identify the key concepts of a field of knowledge, you should ask the following questions: Which are the concepts that are the nodes of the concept net and can be related to many other concepts? Which are the concepts that necessarily keep re-appearing in different contexts when interpreting what you have learnt before? Which are the concepts that arrange specific facts in structures, which contribute to interpreting and apprehending new information and experience? Which are the concepts that - if you are unfamiliar with and unaware of - inhibits you in systematizing various items of knowledge or sensibly utilizing them?" [9] One of the most important of these concepts is the data.

Key words and phrases: key concept, data, primary and secondary school.

ZDM Subject Classification: B32, B33, B72, B73, P52, P53, Q32, Q33.

Being one of the youngest science areas and knowledge areas, informatics is difficult to define. It can be explained by its young age, and on the other hand, by its extremely fast development and its influence on several other knowledge areas.

Although public education in Hungary is provided for 12 years, now we are not considering each age group. This article is dedicated to pupils and students who have to learn (or are supposed to learn) informatics in classes 1 to 10. Regarding 11th and 12th-grade students, only those encounter this concept at a deeper level that are preparing for a career in IT and are participating in programming competitions. We have disregarded this group, just like pupils having taken part in programming competitions at a younger age. Accordingly, we are

now focusing on what an average student needs to know about the data concept and what timing teachers should apply.

1. Fields of Informatics

To define the key concepts of informatics, first you should overview the possible curricula of informatics as a field of knowledge [1, 5, 8, 12, 14].

A. *Algorithmization, data modelling, programming* (At school - and even in your everyday life - you continually execute algorithms, work with data structures e.g. fill in forms and questionnaires, design series of activities and information flow processes, which implies that this world is fully comprehensible only for those who are familiar with the basics of these activities.)

B. *Programming tools* (They include language and other tools which you must be familiar with to be able to implement and try algorithms and data models.)

C. *Performing application tasks with computer* (It means the solubility of everyday problems by using IT tools: video and graphic editing, word processing, spreadsheets, database management and presentation.)

D. *Managing application systems* (Here you should separate the knowledge of applications from the ability to manage rapidly aging devices; although teaching how to use them, of course, should take place parallelly.)

E. *Computer-aided problem solving* (In this topic you should start from a problem emerged - e.g. organising a school trip. First you should approach it as an organisational task, then you should choose the tools and devices for the sub-tasks - not necessarily IT tools -, and then create a new tool if necessary.)

F. *Infocommunication* (You should be aware of the social impact of ICT technologies and adapt to the changes; you should use ICT tools properly.)

G. *Media informatics* (There appeared media heavily infiltrated with IT tools that require IT expertise for proper use; the electronic counterparts of the traditional media open up new opportunities and new media appear; all these might raise cognitive processes and entertainment to another level.)

H. *Informatics tools: Principles of their operation and application* (A wide variety of hardware and software tools are available, the appropriate use of which every user must acquire.)

I. *The mathematical basis for informatics* (The mathematical foundations necessary for IT skills are not included in the mathematics curriculum or they

are not discussed where they are needed - mathematics at secondary school, for instance, does not deal with the matrix, but spreadsheet skills demand the formulation of this concept at least in an informal way - which is alright but it implies that it is the ICT subject that should discuss this issue as well as other applications of mathematics.)

J. *Informatics and society* (It is worth learning about the history of informatics - as it is part of the culture -, dealing with its possible development and its current impact on society, data security, data protection and the ethical issues of applying information technology.)

2. Informatics competences

The key concepts of informatics cannot be separated from informatics competences, and it is necessary that they are in harmony: [6, 11, 12]

- Algorithmic thinking
- Data modelling
- Modelling the real world
- Problem solving
- Communication skills
- Application skills
- Team work, collaboration, interoperability
- Creative skills
- Orientation and information skills
- Systemic thinking

3. Key concepts of informatics

The key concepts of informatics are partly based on more general concepts of other fields (e.g. sign - data are described with signs, signs are used when you communicate, and they are used to describe documents). On the other hand, these concepts appear in other subjects as well (e.g. model, problem). [9, 13]

- algorithm
- data

- document

- communication

- model

- task

- problem

- project

- program

- software

- hardware

Please remember that this overview of key concepts does not mean that it is exactly what should be taught at school. This article is about what teachers need to know so as to teach well! There are often functions related to the concepts: concepts are the ones we work with, while functions are the activities carried out on them. Consequently, it might be interesting to overview the activities.

4. ICT key concept: Data

Every day you fill in forms or prepare some to be filled in by others. The goal of the latter is to obtain or pass on information, which should be supported datasets and data structures. Therefore, it is essential to describe objects of the world by data. [9] In the world there are entities that can be classified into groups - so called types.
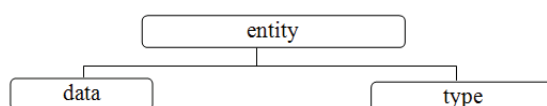

Figure 1. Data as an entity and type

Things can be described if you know and name their properties. It means that after recording the properties considered essential, things are classified according to the quality and quantity of these properties. Data are values expressing the properties of individual objects.
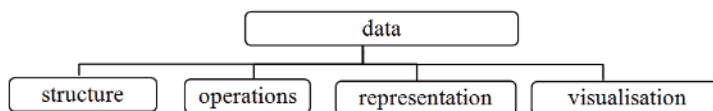
Figure 2. Concepts related to data

The object or device that records (transmits) data are called data transmitters. There exists at least one code system that belongs to a data transmitter, which can be technically displayed on it.

Data are typified according to how and what we would like to use them for. The two most important groups are text data and numeric data.

Data modelling is aimed at revealing data relationships, and developing a skill in students to enable them to consider and study a data processing problem from different points of view. This way a data model is used

- to reveal properties of real world objects necessary for solving tasks;
- to define which properties identify real objects unambiguously;
- to define relationships among the objects represented;
- and set operations that can be performed on the data.

An important question is when you can regard dealing with data as "real" data modelling. In my opinion, from the very beginning! This view of mine is reinforced by an article by Lyv English [3].

A data model contains the following elements:

- entity: a real-world object that can be distinguished from other objects,
- attribute: the characteristics of objects that are used to describe the object,
- relationship: how the objects are related to one another.

One of the concepts, the relationship, is likely to appear later than those of entity and property. In most part of primary and secondary education, obviously, you can only meet the entity-relationship model, from which the relational data model evolves only towards the end of secondary education.

4.1. Areas to be developed in years 1-4 (age group 7-11)

Pupils should realise that everyday data can be classified as numbers, text and other such as col-our, drawing, music etc. In addition, they should learn to distinguish them.
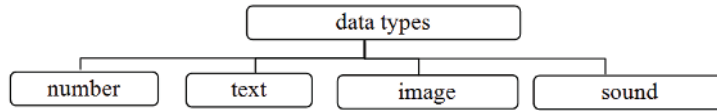
Figure 3. Types of data in years 1-4

This age group can meet two types of data:

- data that are to be processed - then they are related to the concept of algorithm;
- data that are to be displayed - then we get to the concept of document.

It is now that they should understand that objects can be described by values, i.e. one or even several value sets can be assigned to objects (e.g. a flower can have a colour, a name etc.). Moreover, the very same property can be described in several ways (e.g. a month has a name and a sequence number). Some objects can be clearly identified by their properties, while others cannot; e.g. The name of a class is 4A. The colour of the flower is red.
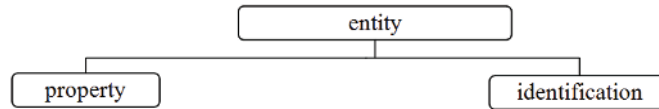
Figure 4. Data characterising entities in years 1-4

Now pupils are to learn the basic concepts of orientation, directions and the measurability of distances. This is mainly achieved by using algorithmic games. For this age group, distance can be interpreted in several ways:
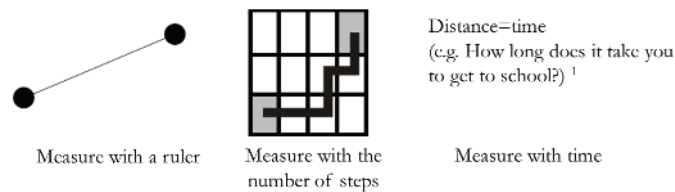

Figure 5. Concept of distance in years 1-4

---

[1]Interestingly, the concept of light year - where distance is measured by time - was introduced in physics much later and with more difficulties.

4.2. Areas to be developed in years 5-6

An important step further would be to assign an operation set to a value set, and to understand that different operations can belong to objects represented by the same values (e.g. two integers can be added together, whereas there is no point in talking about the addition of the ordinal numbers of two months).
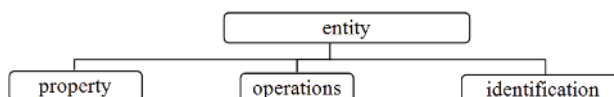

Figure 6. Data characterising entities in years 5-6

Consequently, this is the age group when basic elementary data types, as well as the limitations of their dimensions are clarified. It is important to know, for instance, that - due to the finite value set - the concept of integer in computer science is not identical with the concept of integer in mathematics [7]. The ordinary concept of property slowly becomes type concept.
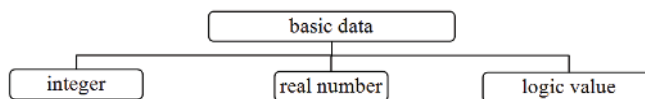

Figure 7. Basic data types in years 5-6

It is worth discussing the means and methods of manual data recording in this age group, as well, since students will encounter tables and graphs in various lessons; one of their first data recording devices is the school report book. They will also meet tables with competition results, or they will have to sort a table by scores; then later they may have to find relations between the tables. Elements of a table can be identified by their place, which means that there appear properties that can be described by structured data - composed data.
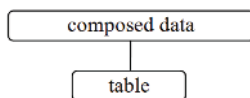

Figure 8. Composed data in years 5-6

The expressive concept structured and unstructured data. Structuring the same data set according to several criteria.

Data can be visualised in a number of ways: by directly defining their value, or by some trans-formation or in a chart. A transformation can be the first step towards separating the value set of data and their visualisation (e.g. when sequence numbers of months are stored, but we want to communicate with names of months).
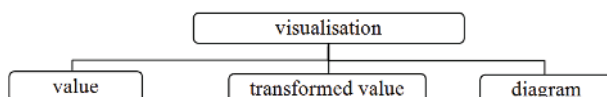

Figure 9. Data visualisation in years 5-6

4.3. Areas to be developed in years 7-8

At this stage we can introduce type structuring (i.e. the method of the constitution of models), so you can talk about composed types. We should raise the awareness of the data concept: the distinction between primary (number, character and Boolean) and composed data (array, table and text)!
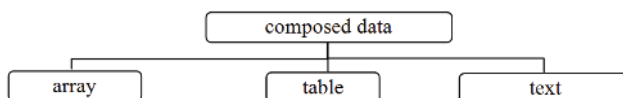

Figure 10. Composed data types in years 7-8

In arrays data are identified by their sequence numbers, whereas in more natural tables, position is not usually important information. There is one important difference: elements of a text (i.e. characters) can be identified by their sequence numbers, but they differ from an array not only by the fixed type of their elements, but by the operations.

Although the range of basic types do not seem to be expanding, a vital novelty needs to be introduced at the conceptual level: types with a limited value set when compared to programming languages (number visualisation by computer), for which, of course, there may exist language tools in certain languages. For example, the number of elements cannot be a negative number; the sequence number of months can range between 1 and 12; the year of birth of people living today cannot be arbitrary; a character string representing people's names can contain only letters, dashes and spaces etc.

We cannot avoid dealing with input and output data necessary for information management, as well as with their assigning. (At this important stage of abstract thinking, students are able to separate and handle details of specific activities independent of the purpose of the activity, i.e. they are capable of operating with these activities as "black boxes".) They can make an analysis from a given point of view, and then utilize the results for a particular purpose. When processing data, students are challenged to enter data describing real-world entities into the computer, making the computer calculate new data from the ones provided, and then display data for the outside world. [10]
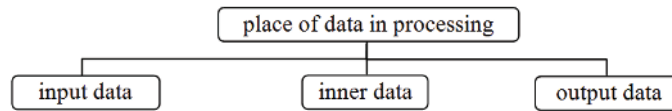

Figure 11. Data in the computer and the outside world in years 7-8

Regarding data management, it can be stated that the value of some data is permanent, while others can be changed or calculated. Here we may note about the concept of algorithm that computer algorithms use data to calculate some other data i.e. the execution of a computer algorithm is always some kind of calculation.)
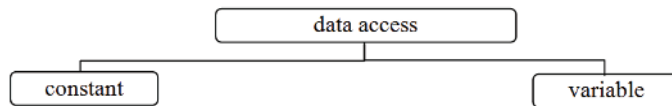

Figure 12. Data access in years 7-8

Various methods of data visualisation may appear in various applications: the data themselves are listed, the data layout or highlighting are given, the data are shown in a diagram (e.g. if you were on holiday from 10th to 27th July, you can visualise the two dates, you can colour the specific days of July in the calendar, or you can plot a sub-interval of the month as an interval, when you were away etc.).
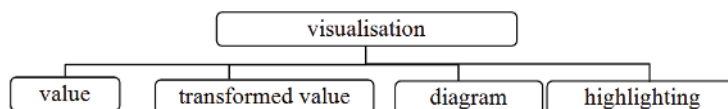
Figure 13. Data visualisation in years 7-8

4.4. Areas to be developed in years 9-10

In course materials for this age group, there may appear some basic types the value set of which we set ourselves. Naturally, you may want, for instance, to identify months by their names and not by their sequence number; or to define colours by their names instead of their numeric codes - these are all texts the value set of which is limited and can be listed.
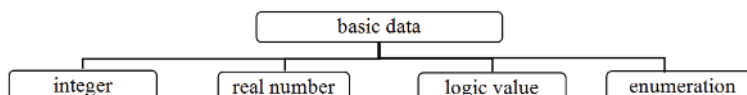
Figure 14. Basic data types in years 9-10

A novelty for the age group can be to learn that types can arbitrarily be defined i.e. their structure and operations can be determined, and they can even be named. The best ones (probably those interested in an IT career) could meet type abstraction and the concept of type hierarchy when a type is built up (i.e. integrated) from other types (traditional type definition), or built on other types (object-oriented type derivation).
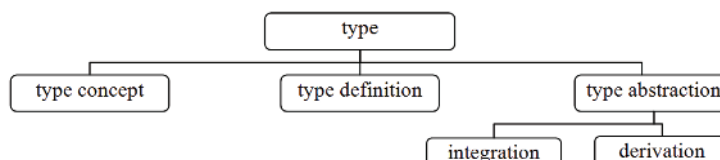
Figure 15. The concept of type in years 9-10

Another important novelty is the separation of the visualisation of types and data.

Basic and composed data (array, record, table, set and file) and file types (sequential and non-sequential). The elements of a set are not related; the sequence-like ones can be identified by their place occupied in the sequence.
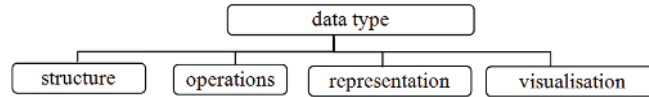
Figure 16. Concepts related to data in years 9-10

At this stage, data structures existing in the world of algorithms, like stacks, queues, trees and graphs, are simply arrays or sets (e.g. a line is an array that you fill in continuously from the beginning, and you also keep record of where you are in the processing).
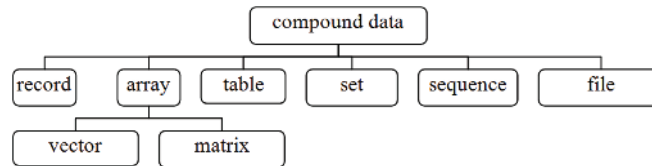

Figure 17. Composed data in years 9-10

Students should be aware of the basic rules of data modelling, and understand that a database is not a simple file, but a pre-planned, grouped system of data and their relationships. You can highlight some properties of objects that clearly identified the entities (key). A key could be an artificial sequence number-like something, a natural primary value (like peoples name if there are not two identical names), and composed value (for instance, everybody is likely to be clearly identified by their name, address and date of birth).

One purpose of data processing is to make the data loaded in some form available and accessible for others to perform queries on them from other points of view. Owing to exploring data relations, data query is a creative-analyzing process. Here data can be classified into several groups like tables, sets etc. with various relations among them, which also need studying.
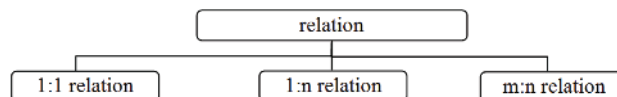

Figure 18. Data relationships in years 9-10

You can, of course, raise the question whether all members of a group are involved in a relationship, or whether the members of a group can be related to one another (recursive relationship).

Modelling also involves knowing about what information the program will be able to provide when ready. So it is necessary to understand what kind of information can be obtained from the data i.e. you should ask the following important questions: *What do we need?, What shall we ask?, What can we find somewhere else?* etc., which means that the most difficult task is to raise good questions. This is what will help you to compile good questionnaires, to define the scope of input data, and decide on the scope and visualisation of the results.

Regarding data - often data found on the Internet, but databases, as well -, there comes up a common problem: that of access rights. Some access rights can be limited in time: e.g. several people can watch the same data at the same time, but there is only one person that can make changes (while others cannot even watch them).
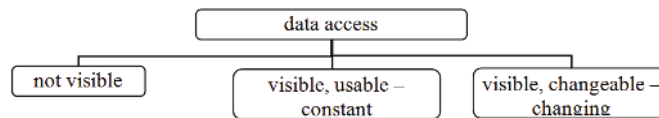

Figure 19. Access rights in years 9-10

Access can be provided to the whole data or some parts of composed data such as tables in databases, an entity, a property etc. Access can be tied to entitlement, e.g. to an identifying password (or even to a hierarchical identifier system with a fixed hierarchy of rights), or to the payment of a fee.

Using a spreadsheet, a wider application of tables and graphs generated from them; calculation and visualisation of simple statistical data; evaluation of physical-chemical measurements. This is where you can point to an interesting link with the world of programming languages, namely with functional languages: functions can be defined on certain domains of the tables although students might have difficulties with recursive calculations. [4]

Visualisation could be expanded by spatial placement i.e. a map - the result of a search can be a location on a map or the map itself.
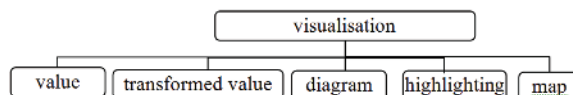
Figure 20. Visualisation of data in years 9-10

Now the concept of data is again strongly related to the concept of algorithm. Data can be classified according to the their place in processing (global, local or private).
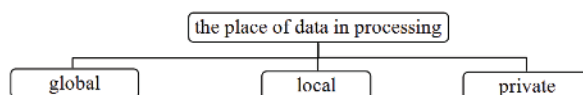

Figure 21. Data processing in years 9-10

5. Why can the identification of key concepts play an important role in planning the teaching-learning process?

"When designing the knowledge to be acquired in a field of science or in a subject, it is essential to see the wood for the trees. We need to be aware of which of the many elements of knowledge or concepts to be learned are really important. A possible way of weighing the content is to try and identify the key concepts for a field of science first. If it is done, you should select those that must be taught in primary and secondary education by all means." [9]

After the algorithm, data is probably the second most important concept in computer science. These two concepts are inseparable, and therefore, they should be taught together. On the other hand, data is closely related to the third basic concept, the document, as well, since documents more and more function as data, a priority area of which is the world of split, text databases. [2]

The uniform structure of the data concept could drastically re-structure the teaching-learning process: those topics of the traditionally independently taught programming, spreadsheets, database management that are associated with data could be grouped around a common theme.

Many other issues may arise which belong to tertiary IT education or to secondary education focusing on preparing students for an IT career: data - information, data - communication, data visualisation on the computer, effective data visualisation, data - programming language: XML, SVG.

References

[1] Informatika a Nemzeti Alaptantervben (Informatics in the "National Ground Curriculum"), *EVilág (eWorld)* **10** (2003), 22–25.

[2] L. Menyhárt, Can a language be before "the first programming language"?, *Teaching Mathematics and Computer Science* **9**, no. 2 (2011), 209–224.

[3] L. English, Data modelling in the beginning school years, *Mathematics: traditions and [new] practices*, AAMT & MERGA, 2011, 226–234.

[4] P. Hubwieser, Functional Modelling in Secondary Schools Using Spreadsheets, *Education and Information Technologies* **9**, no. 2 (June 2004), 175–183.

[5] P. Szlávi and L. Zsakó, Informatics as a Particular Field of Education, *Teaching Mathematics and Computer Science* **3**, no. 1 (2005), 151–162.

[6] P. Szlávi and L. Zsakó, Informatikai kompetenciák: Algoritmikus gondolkodás, INFODIDACT 2010 - 3. Informatika Szakmódszertani Konferencia, Szombathely, Hungary, 2010. április 22-23., 2010, konferencia-CD.

[7] T. Teodosiev and A. Nachev, Some Pitfalls in Introductory Programming Courses, *Informatics in Education* **11**, no. 2, Vilnius University (2012), 241–255.

[8] The National Curriculum for England, Information and Communication Technology, Qualifications and Curriculum Authority, London, 1999,
`http://www.nc.uk.net/download/IKT.doc aktuális online NAT.`

[9] NAT-hoz illeszkedő kulcsfogalom-rendszer, kulcskompetencia-térkép, tanulmány, (V. Vass, ed.), Nemzeti Tankönyvkiadó, Budapest, 2011.

[10] V. Dagdilelis, M. Satratzemi and G. Evangelidis, Introducing Secondary Education Students to Algorithms and Programming, *Education and Information Technologies* **9**, no. 2 (2004), 159–173.

[11] L. Zsakó, Z. Kátai and K. Nyakóné-Juhász, ICT methodology, *Teaching Mathematics and Computer Science - Infodidact* (2008), 3–24.

[12] L. Zsakó, Fejezetek az informatika szakmódszertanából, *Habilitációs értekezés*, Debrecen, 2007.

[13] L. Zsakó, Az informatika kulcsfogalmai: algoritmus, Informatika a felsőoktatásban 2011, (Debrecen, 2011. augusztus 24-26), Debreceni Universitas, 2011, konferencia-CD.

[14] L. Zsakó, Teaching Informatics in Hungary, *The IOI'96 NewsLetter* **2, 3, 4**, Neumann János Számítógép-tudományi Társaság, Budapest (1995), 5–6.

PÉTER SZLÁVI, LÁSZLÓ ZSAKÓ
FACULTY OF INFORMATICS
EÖTVÖS LORÁND UNIVERSITY
HUNGARY

E-mail: `szlavip@elte.hu`

E-mail: `zsako@caesar.elte.hu`