education sciences

MDPI

Systematic Review

Computational Literacy as an Important Element of a Digitized Science Teacher Education—A Systematic Review of Curriculum Patterns in Physics Teacher Education Degrees in Germany

Daniel Braun [1,2,*] **and Johannes Huwer** [1,2,*]

[1] Chair of Science Education, University of Konstanz, 78464 Konstanz, Germany
[2] Chair of Science Education, Thurgau University of Education, 8280 Kreuzlingen, Switzerland
[*] Correspondence: d.braun@uni-konstanz.de (D.B.); johannes.huwer@uni-konstanz.de (J.H.)

Abstract: Computational literacy (CL) has become indispensable for teachers and learners as part of 21st-century skills. Therefore, corresponding models for teacher education are being further developed internationally from a scientific perspective. In parallel, content and competencies are being enhanced in the respective subjects at the curricular level of teacher training. In this context, we consider it important to examine the current status of this development. Since, to our knowledge, there are no comparable scientific studies, we have taken Germany as a representative example of the international education system and systematically analyzed the module handbooks of the physics teacher training courses at methodically selected universities. For this analysis, we used three research questions focusing on CL: In which physics content does CL play a role? Which computer science competencies or knowledge can be identified or derived? Are they described implicitly or explicitly? Our results suggest that CL is integrated very differently in terms of quantity and depth of content among the universities we examined. For example, there is often a very strong focus on computer-based data acquisition, but few programs also have specialized courses addressing CL more explicitly or integrate additional computer science competencies. CL is primarily taught in laboratory courses and frequently in subject-didactic courses. Nevertheless, the depictions presented in the purely subject-oriented and basic lectures lack specific computational literacy skills or knowledge. Furthermore, the fact that many programs only offer implicit descriptions of CL skills indicates that the integration of these skills has not progressed very far in practice.

Keywords: computational literacy; digitalization; STEM; science education; science teacher education; teacher training; computer science

Citation: Braun, D.; Huwer, J. Computational Literacy as an Important Element of a Digitized Science Teacher Education—A Systematic Review of Curriculum Patterns in Physics Teacher Education Degrees in Germany. *Educ. Sci.* **2023**, *13*, 1063. https://doi.org/10.3390/educsci13101063

Academic Editors: Magdalena Ramos Navas-Parejo, Pedro Antonio García-Tudela, Lucía Lomba Portela, Marco Antonio Zamora Antuñano and Eleanor Dommett

Received: 25 August 2023
Revised: 14 October 2023
Accepted: 19 October 2023
Published: 23 October 2023

1. Introduction

Digitization on the societal level represents a major challenge for Germany. To be innovative and remain competitive in the future, digitization must be integrated into the education of various disciplines. This goal and the associated challenges are international and, therefore, affect all industrialized countries [1–3]. Many content-related issues in education are equally relevant in other countries, and (regional) governments must also adapt teacher training programs and competence descriptions [4,5]. Because education and teacher training in Germany are also based on the international system consisting of political decision-makers, universities, and subject-based teaching with different educational pathways, we take a closer look at K12 teacher training at universities in Germany as an example in this paper. Due to the federal landscape in Germany in educational matters, we can look at up to 16 different implementation systems, where, fortunately, the common basic understanding of education favors a comparative analysis. For this purpose, we explicitly look at module manuals for teacher training programs in physics. This particular research approach is also found in the literature for comparable research settings [6–8]. Our guiding question is how computational literacy is integrated as an important basic framework for sustainable teaching and a contemporary understanding of physics subject content. The aim of this review is to systematically analyze how computational literacy is already embedded in physics teacher education via module handbooks. These module handbooks are particularly well suited for this analysis because, on the one hand, they are independent of an individual instructor's subject focus and have a lasting impact or binding force over several years. On the other hand, their objectives and structure make them relatively easy to compare, and they are freely accessible on the Internet. Thus, we can systematically analyze similarities and differences and capture the state of physics teacher education today more accurately. This type of analysis is important because, to the best of our knowledge, such data have not yet been collected for this purpose. This is also why we limit ourselves to one subject and one country to gain first insights; in this way, we can focus on the different implementations of computational literacy. We consider differences due to different nationalities and broader base sources to be meaningful only after we have developed a more concrete frame of reference with the present research questions. Although we explicitly use Germany as a reference, our transparent and systematic analysis enables transferability to other countries with comparable framework conditions because the substance of physics education does not change at national borders. Alternatively, valid conclusions can still be drawn by contrast if some conditions differ fundamentally.

An important starting point is that (science) education in schools in Germany is usually provided separately by subject. Some types of schools may practice interdisciplinary teaching (e.g., "Gesamtschulen", comprehensive schools). However, this type of teaching is not widely available, so we do not include it in our study. More commonly, there is no integrative STEM teaching, but rather the subjects physics, chemistry, biology, and computer science. Accordingly, teacher training at universities is also strictly separated by subject. On the one hand, this can be a limiting factor in transferring our findings to other educational landscapes. However, on the other hand, it also facilitates the systematic search for targeted subject-specific aspects. In this way, domain-specific aspects such as subject knowledge or attitudes can be examined more closely [9–11]. It also allows new learning approaches and concepts to be derived in a more goal-oriented way based on the subject perspective in the context of digitization [12]. Even in an integrative STEM class, there is, for example, physics content that calls for specific computational literacy.

Two disciplines, science and computational literacy, already meet at this point, independent of the instructional preparation. However, if another subject (e.g., chemistry or chemical physics) were to be analyzed along with these already implicitly related subjects, then relevant findings could be overlooked, or undesirable correlations could distort the analysis. Therefore, we argue that this subject-separated view is ultimately helpful for our guiding question, and yet a transfer of our findings to integrative settings is also reasonably possible.

Science teaching in the 21st century must integrate digital competencies appropriately [13–15], both from the perspective of subject didactics (cf. [16]) and from the perspective of subject content (cf. mathematics [17,18]). In this context, it is important that these competencies can be built up sustainably (cf. possible issues or obstacles [19–21]), for example, to prevent future teachers from merely acquiring product knowledge (PKN) about digital tools, which inevitably becomes obsolete. Therefore, it is helpful to consider computational literacy as independent and originating from the subject discipline of computer science. This is because this subject distinction allows module handbooks to be analyzed from two perspectives: What competencies can computer science provide to improve and develop science teacher education sustainably? What science content requires or benefits from computer science competencies? With the following research questions (see Section 3), we would like to address this ambivalence to highlight initial goal-oriented similarities and differences systematically and to raise possible further questions.

2. Theoretical Background

As described above, digital education and the associated competencies are central challenges across national borders in the 21st century. For this reason, the European Commission responds to this challenge with its "European Framework for the Digital Competence of Educators (DigCompEdu)" [22], which describes a competence framework that addresses the digital competencies of teachers. For instance, competence requirements or possible training programs are explained, which can be taken up by member states.

2.1. DPACK Model and DiKoLAN as Reference Points

On a more fundamental level, the DPACK framework (see Figure 1) describes teaching and learning in a digitized society separated by the domains of digital knowledge (DK), pedagogical knowledge (PK), and content knowledge (CK). This DPACK model [23,24] further develops the TPACK framework [25,26]. Thyssen et al. argue that digitization and teaching and learning in today's society are necessarily embedded in social and cultural knowledge (SK) and should, therefore, be considered accordingly in an interconnection with technological knowledge (TK), PC, and CK. Combining SK and TK results with DK is especially relevant for teacher training. Thus, school education can help build computational literacy skills that enable students to actively participate in sociocultural and societal life. Transferred to the topic discussed in this paper, this raises the question of which computational literacy skills are needed at the intersection between digitality, as described above, and content expertise beyond the mere technical perspective so that learning STEM education remains contemporary. Due to rapid technological development, it is obvious that computational literacy is increasingly needed to conduct subject-specific teaching—especially in STEM subjects—in certain subject areas or contexts; for example, Benz and Ludwig [27] discuss digital technologies and media use in physics classrooms, emphasizing the need for specific competency frameworks for handling subject-specific technologies like digital data acquisition systems. Therefore, certain computational literacy competencies are implicitly necessary to reach such goals, but they must also be strongly subject-oriented. Concerning the TPACK model, the further division of the intersection TCK by the DPACK model is particularly interesting: while, e.g., pure product knowledge or the unreflected use of digital tools can correspond to TCK, we can distinguish, in the evolved division DCK, computational literacy skills in the sense of sustainable concept knowledge from TCK. Suppose that we now combine the pedagogical dimension as well. In that case, in the intersection of DPACK, these basic computer science competencies concern subject-didactic and subject-independent teaching–learning aspects. Following this model, our analysis of the module handbooks can also provide information on how computational literacy is integrated into the physics degree programs under consideration as part of TCK, DCK, or even DPACK. Therefore, we also differentiate whether the module handbooks focus on these competencies in subject-specific or subject-didactic courses. As (prospective) teachers, they often want to adapt to the digital world but often lack sufficient computational literacy, especially with tools like augmented reality. Freese et al. [28] show how tools and DPACK can help physics teachers improve their digital and modeling competencies for AR experiments to redesign modeling more creatively as an essential part of science education for learners in a more accessible approach. So, computational literacy skills improve teachers' possibilities of didactically rethinking essential aspects of science education.

On a practical level, the orientation framework "Digital Competencies for Teaching in Science Education" (DiKoLAN, title translated in English) exists in Germany, Switzerland, and Austria [29]. Becker et al. [30] describe the basic digital competencies that teacher students in the natural sciences should possess after completing their studies. Because many universities in Germany have voluntarily committed themselves to implementing DiKoLAN, we can use it as an overarching blueprint for the module handbooks and analyze the immanent computational literacy as an example. This gives us a first indication of which computational literacy skills are consensual from the perspective of science course design in the DACH higher-education region and should be included in the module handbooks (and courses) accordingly [31]. On closer inspection, however, it is also apparent that the DIKOLAN framework has certain weaknesses and gaps. For example, there is no adequate reflection on the use of digital media or the concrete inclusion of computational literacy. In addition, there is a lack of explicitly mentioned relevant computational literacy skills to consider digital development in natural science subjects and their teaching. If strictly implemented, this could manifest in module manuals.

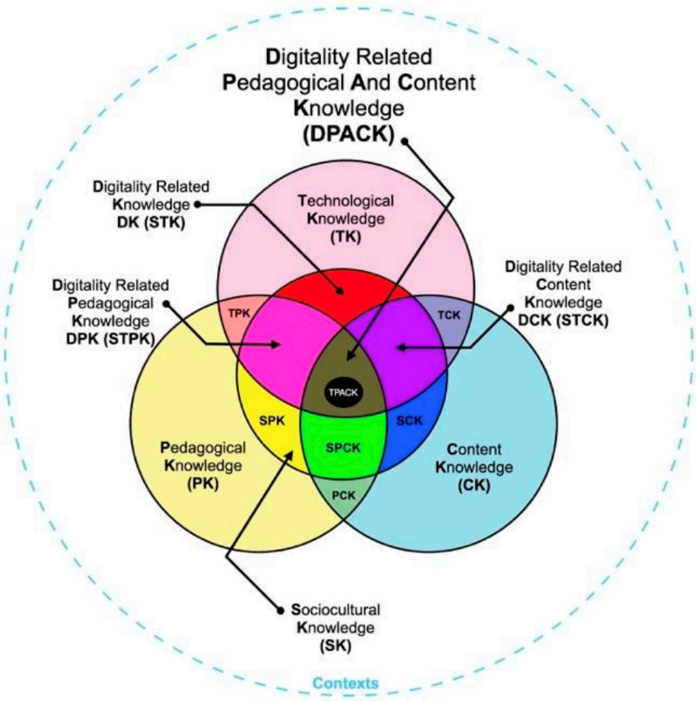

Figure 1. Visualization of the relevant elements of the DPACK framework according to Thyssen, Huwer, Schaal, and Irion [23].

Regarding such gaps, there are already contributions based on the TPACK model, for example, which attempt to assess such gaps and limitations more precisely and to investigate possible solutions [32]. Henne et al. [33] showed how to implement Digital Competencies in University Science Education Seminars Following the DiKoLAN Framework. For example, the competence expectations in the area of measurement value and data acquisition include the thematization of digital systems (e.g., computer-supported measurement value acquisition) as an access route to (digital) basic competencies, whereby, at the same time, the current requirements of subject-specific research should also be considered. From a conceptual point of view, it is thus about the effect and role of computer science systems as a conscious part of the measurement process, e.g., considering the input–process–output (IPO) model or as an aspect of the automation of measurement processes.

Comparably, the guiding principles of data and encoding play a role in the competence area of data processing, as it is a matter of converting data from one encoding into another. Data types and structures are also taken up in the question of the advantages and disadvantages of different file formats or forms of storage and become more relevant, especially when dealing with large amounts of data. Suppose that these approaches and principles are specifically taken up holistically under computational literacy. In that case, the module manuals TCK and DCK cover them adequately in the DPACK model. If the focus is only on technical aids, only the TCK part appears as part of TPACK. The difference in these competence categories is computational literacy. It separates teachers into two groups: some use tools without a sustainable learning effect, and others create such adaptive learning settings, e.g., by adapting a learning platform to create identification keys for various living species [34]. The latter shows the role of digital tools in fostering collaboration, cooperation, and participation. It also highlights the need for teaching computational literacy to enable sustainable learning in digital contexts. If these competencies are explicitly missing, then the question remains whether the necessary computer science competencies are nevertheless considered in the degree program in an alternative form.

In summary, politically and socially formulated target models and requirements of contemporary subject-specific and educational science training show the need to effectively integrate computational literacy into STEM education. Based on these considerations, the module handbooks should include the corresponding competencies and knowledge.

2.2. Linguistic Aspects

Language, especially subject language, is crucial in acquiring competencies. This is especially true in STEM education. For this reason, it is problematic that terms are not used consistently for digital or information literacy in the literature. For example, Braun and Huwer [35] indicate no common terminology for describing computational literacy in the scientific literature, especially in the STEM education field. Thus, information literacy [36], data literacy, digital literacy, computational thinking [37–43], and even computer science (competencies) are used inconsistently in the literature. These terms may be equated or understood contradictorily.

For this paper, we must keep in mind that we work with module handbooks in the German language and that in the German educational language area, e.g., the language term "informatics competencies" ("Informatische Kompetenzen" in German) is widely used. However, in the international context, it is translated differently by the above terms [35,44,45]. For these reasons, we use the following terminology in this paper: we understand computational literacy as an overarching term that summarizes the knowledge, competencies, and concepts required in the 21st century from the subject of computer science [46–50]. We divide computational literacy into three basic concepts of computer science: automation, digitization, and computer systems [51]. Subordinated to these, computer science competencies and knowledge describe concrete aspects of these basic concepts, which prospective teachers can acquire, e.g., implementing programming concepts or knowing the basics of data encoding.

In summary, we focus on science competencies and knowledge that can play a crucial role in science education in combination with computational literacy. However, we exclude purely user-knowledge-oriented use of digital media in teaching–learning scenarios. We also exclude competencies from the areas of media education and media criticism that are often wrongly associated with computer science. Thus, the focus is on competencies that take up relevant computer science concepts or content. In this respect, we will include all content and competencies that address computer science in a narrower sense, even if the source documents do not explicitly describe them as computational literacy skills. In addition, computational literacy emphasizes that, especially in science teaching–learning scenarios, the computer is used both as a tool and as an object of instruction, resulting in new competence expectations for teachers (see Section 2.1).

2.3. Narrowing the Scope

Based on the concept of extending the TPACK model to the DPACK model, we focus our research concern on the fact that computational literacy must be fundamentally integrated into the education of physics teachers because both the methodological (TPCK) and the content-related (CK) competencies have to be fundamentally developed in the context of the digital transformation. In doing so, we are concerned with the essence of natural science, which can no longer be adequately covered without this addition of computational literacy. Otherwise, future teachers will no longer be able to understand essential physics aspects or prepare them in a didactically appropriate way. We must emphasize that we do not focus on a value-added analysis of TPCK competencies, as described, e.g., by the SAMR model [52]. Here, we present how teaching and learning change because of digitality and technology.

In contrast, the SAMR model can be used to explain how digital media can improve the design of teaching and learning settings. However, this is a different approach. We focus on the fundamental integration at the curricular level (based on the module handbooks) from the perspective of the theory-driven framework models such as DPACK.

Furthermore, we derive the necessity of integrating computational literacy regarding curricula design because, from our point of view, physics is incomplete. Therefore, we are not interested in showing the integration of TPCK as a toolbox with the user reference as part of a value-added discussion with the students. However, before the students come into the lecture hall, the curriculum should already be fixed, including which computational literacy aspects have become indispensable and must be used. We think here, for example, of the modeling and simulation of physical theories with the help of computational literacy. Especially in astrophysics, quantum physics, or quantum computing, physics teachers without computational literacy can remain in the user perspective if they cannot teach creatively or actively design due to a lack of computational literacy. We can also draw an analogy to mathematics, where the added value of elementary mathematical competencies is no longer discussed in physics courses. This fact has long been reflected in the module manuals of physics courses, especially in teacher training courses, because otherwise, teachers cannot teach physics. Therefore, without narrowing our focus to TPCK in the above sense, our focus is valid for teacher education degrees.

Beyond these considerations, we know that prospective teachers have limited resources to acquire additional module content in supposedly unrelated areas due to their workloads. However, our research interest is understandably justified from the systematic perspective of developing science courses in the 21st-century model-guided way. This is because it allows us to determine which aspects of computational literacy need to be integrated, in which way (integrated or stand-alone modules), and in which scientific subject areas they are already considered necessary. Moreover, we maintain that, in turn, computational literacy alone may not be sufficient to achieve these goals, but the extent to which it is already a key component is what our paper aims to demonstrate.

3. Research Questions

For the systematic analysis of the module handbooks of selected physics courses, we target the following research questions:

RQ1: In which physics content or modules does computational literacy play a role?

In this question, we consider computer science as a subsidiary discipline that functions in a supportive way from the physics perspective, just as mathematics does, for example. In addition, this also allows us to investigate whether courses dealing with computational literacy are more likely part of the subject science or subject-didactic lessons.

RQ2: Which computer science competencies or knowledge can be identified or derived?

This question is about identifying existing sub-aspects of computational literacy from the perspective of computer science as a subject discipline.

RQ3: Are computer science competencies or knowledge described implicitly or explicitly?

This question focuses on two issues: On the one hand, we investigate whether computational literacy is explicitly described within a module or whether we can assume this based on the physical aspects and the depth of representation in the module handbook. On the other hand, we consider whether computational literacy is spun off into independent (sub-)modules or embedded in canonical courses such as canonical lectures. For the spin-off, it matters whether the predominant part of the module focuses on computational literacy.

4. Methodology

We followed the PRISMA standard [53] for the systematic review to compile the qualified sample of module handbooks based on valid criteria. We illustrate this selection process in Figure 2. Therefore, we first selected up to five universities per federal state that are not too specialized in their range of study programs due to their general profiles. For example, we excluded medical universities and the University of the Federal Armed Forces. We also did not consider universities of applied sciences or universities without relevant courses for teacher training. This is because our focus is on teacher training programs that qualify teachers in physics for K12 education (German: "Sekundarstufe 2"). If a state has more than five potential qualifying universities, we selected those that have the most students by total number of students in that state. This is because the 16 federal states are responsible for education in Germany's federal structure. In this way, we cover the universities per federal state that, in relative terms, also educate the most students in the teaching profession. To do this, we evaluated GENESIS-Online (the database of the Federal Statistical Office). We used the winter term in 2021/2022, as that was the most recent entry in the database. In the resulting list, we deleted all remaining universities that do not offer a teaching degree program in physics, as already mentioned. If there were any other qualified universities in the state after this exclusion, we did not include them because we wanted to preserve the ranking structure by the relative number of students to maintain comparability from the perspective of reach in teacher education in the respective state—even if we have fallen below the number of 5. We also excluded universities that, according to the database of the German Rectors' Conference (German: "Hochschulrektorenkonferenz", hochschulkompass.de), in which the study programs of the German universities are officially systematically collected, do not offer qualified teacher education programs. Finally, we still had to exclude a small number of universities because, at the time of the survey, the module handbooks were not publicly available on the Internet. More details are described in Section 5.

In the module manuals of qualified universities, we have only considered courses or modules that are explicitly described therein. A mere reference to offerings of other subjects or institutes in the context of elective options we do not classify as sufficiently binding, and therefore, we do not consider it part of the education. If relevant modules in the elective or optional area included computer science competencies, we nevertheless included them if the modules were explicitly described as part of the physics course and a connection in content was recognizable. In addition, these modules had to be either from the physics department itself or specifically designed for students of the physics department. This is because, as a general rule, the physics department can hardly influence the design of courses offered by another independent department if they are primarily designed for students of that other department. Indeed, our fundamental research interest is the integration of computational literacy into science education. Although the individual student's elective education may vary widely, this determination nevertheless satisfies a systematic view of which computational literacy skills are present in the course independently of this fact.

In order to increase comparability, we did not consider the module handbooks in their entirety but only examined two items per course: the description of learning objectives or competencies and the description of contents. This is because we assume that statements in preambles, for example, remain too non-binding without a concrete reference to a course and are even less likely to be taken up with certainty. Additionally, on the one hand, these two items provide the essential information for our research questions. On the other hand, they are contained in all module handbooks, whereas the presence and level of detail of other textual components, such as a preamble or overarching statements, are very inconsistently represented and often do not refer to specific courses but rather resemble certain statements of intent.

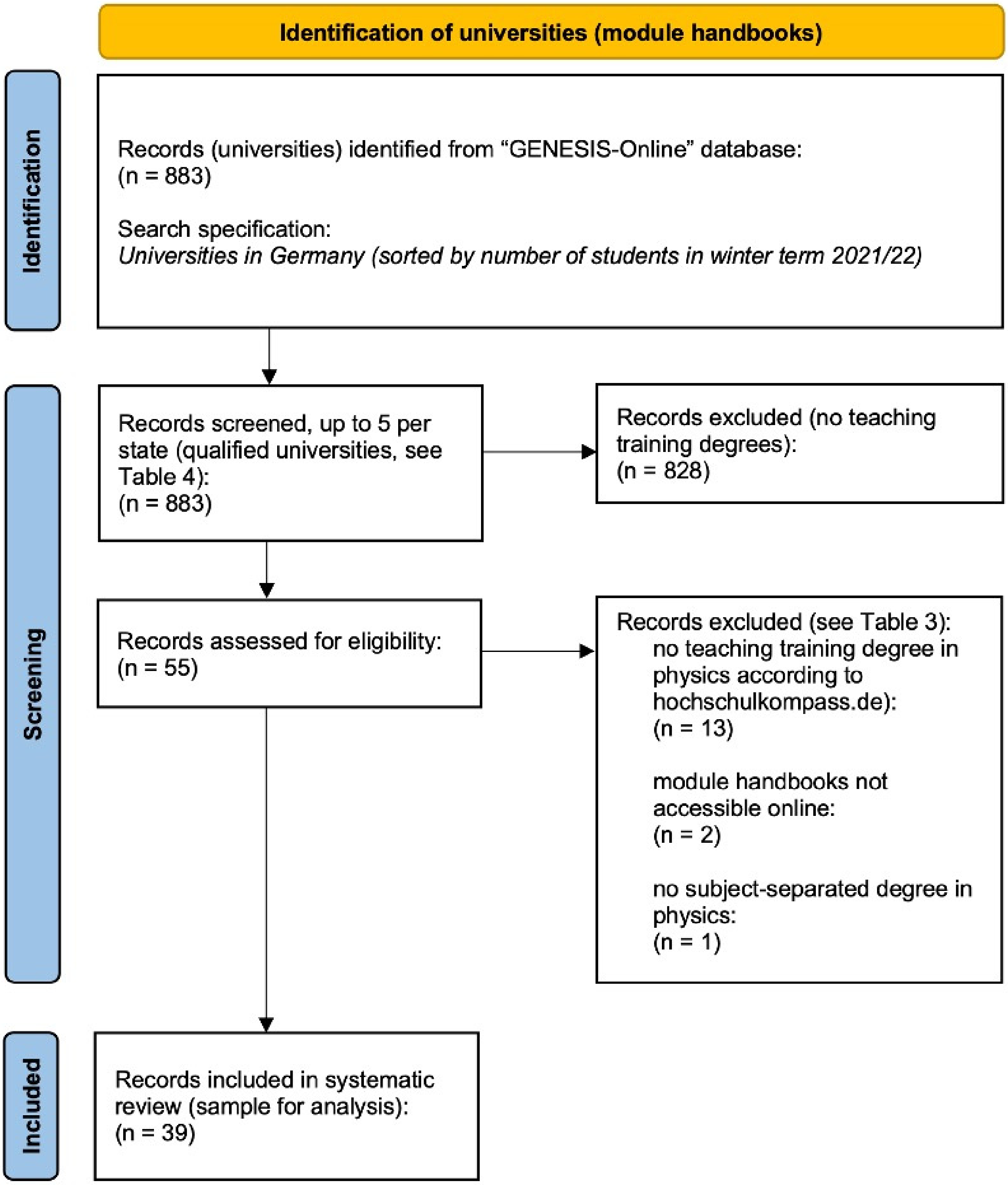

Figure 2. Visualization of the selection process according to the PRISMA standard.

We used the keywords and auxiliary criteria in Table 1 to identify computational literacy skills. The basis of the analysis was German terms and expressions. We have provided the most appropriate English translations to facilitate readability for international readers. Therefore, we adapted relevant and fundamental aspects of computational literacy as keywords, with a special focus on relevance in science education. As part of the working process, we adaptively expanded the list with other relevant terms in the context of the defined keywords in the modules. This allowed us to ensure that different terms and phrases could be considered for the same competence description. Some keywords, such as "digital media", are very weak search terms, so we additionally analyzed the context using auxiliary criteria to assess relevance from a computational literacy perspective.

Table 1. Keywords and other criteria.

German Keywords (in Groups)	English Keywords (Translated)	Auxiliary Criteria or Explanation
*Computer *, *Rechner *, PC- *	*Computer, PC- *	In connection with specific computer science or physics competencies/goal descriptions
Hardware, Informationstechnik	Hardware, information technology	In connection with specific computer science or physics competencies/goal descriptions
Messtechnik, *Elektronik *, Sensoren, Messwerterfassung	Measurement technology, electronics, sensor, measurement data acquisition	In connection with specific computer science or physics competencies/goal descriptions
Daten *	Data *	In connection with concrete computer science competencies/goal descriptions
Numer *	Num *	Numerical. In connection with computer science concepts or questions. Differentiation from pure numerical mathematics
Simulation *, Animation *, interaktiv	Simulation *, animation *, interactive	In connection with concrete computer science application concepts
Software, Code	Software, code	Conceptual theming of programs or independent programming
*Programm *, Implementier *	*Program *, implement *	Conceptual theming of programs or independent programming
Digital *, Digitalisierung	Digital *, digitization	Weak criterion. Connection with computer science concepts or contexts necessary (e.g., digital circuits)
Moderne * Medien, Smartphone, Tablet, App	Modern media, smartphone, tablet, app	Very weak criterion. Connection with computer science concepts or contexts mandatory (e.g., digital circuits). Differentiation from media pedagogy/media didactics/product knowledge
Informatik	Informatics	Inclusion or reference as an auxiliary science or as interdisciplinary mediation

Note: In Table 1, an asterisk (*) stands for any number of letters. Upper and lower case are not considered.

Independently of filtering based on keywords, we also considered modules if concrete computer science contents, concepts, or competencies are mentioned (see Table 2). However, we could not use these terms exclusively because subject matter experts in physics wrote the module manuals from the physics perspective. This is because, at least at the time of the analysis, we assume that the module handbooks or the drafting committees of the physics departments do not (yet) elaborate the computer science terminology and related terms to a relevant extent. However, these aspects of computational literacy nevertheless already occur under other terms. We have incorporated the "Reference Framework for Computer Science" by Röhner et al. [51] in defining these terms.

Table 2. Relevant terms or content-related sub-aspects of computer science terminology.

Basic Concept	Relevant Terms or Content-Related Sub-Aspects
Automation (AUT)	(Implementation of) algorithms, programming skills (incl. naming of concrete programming languages), programming concepts and approaches (iterative, recursive, imperative, declarative, object-oriented), modeling
Digitization (DIG)	Data and coding, data structures, input–processing–output principle, (quantum) cryptography, concepts of operating systems, computer accuracy
Information Technology Systems (CSS)	Databases, data acquisition systems as data-processing systems, technical data processing, microprocessors, use of numerical methods with computer algebra systems

5. Description of the Sample

In this section, we provide an overview of our sample. The details of this selection process are given in Tables 3 and 4.

Table 3. The selection process for qualified universities.

Number of qualified universities	55
Number of excluded universities according to hochschulkompass.de	13
Number of excluded universities due to the lack of free online availability of module handbooks	2
Number of excluded universities due to the lack of a subject-separated physics degree program	1
Number of universities evaluated	39

Table 4. List of qualified universities.

No.	University (GENESIS Name)	Federal State with Rank	Exclusion Criterion
1	Freie Universität Berlin	BERLIN-1 (BE)	
2	Humboldt-Universität Berlin	BERLIN-2 (BE)	
	Technische Universität Berlin	BERLIN-3 (BE)	hochschulkompass.de
3	Universität Potsdam	BRANDENBURG-1 (BB)	
	Brandenb. TU, Cottbus-Senftenberg in Cottbus	BRANDENBURG-2 (BB)	hochschulkompass.de
	Europa-Universität Viadrina Frankfurt (Oder)	BRANDENBURG-3 (BB)	hochschulkompass.de
4	Universität Bremen	BREMEN-1 (BR)	
5	Universität Tübingen	BADEN-WÜRTTEMBERG-1 (BW)	
6	Universität Heidelberg in Heidelberg	BADEN-WÜRTTEMBERG-2 (BW)	
7	Universität Freiburg im Breisgau	BADEN-WÜRTTEMBERG-3 (BW)	
8	Universität Stuttgart	BADEN-WÜRTTEMBERG-4 (BW)	
9	Karlsruher Institut für Technologie, Karlsruhe (U)	BADEN-WÜRTTEMBERG-5 (BW)	
	Universität München in München	BAYERN-1 (BY)	no module manual available online
	U Erlangen-Nürnberg in Erlangen (siehe HS1310)	BAYERN-2 (BY)	no module manual available online
10	Universität Würzburg	BAYERN-3 (BY)	
	TU München in München (2001–2016 HS1630)	BAYERN-4 (BY)	lack of a subject-separated physics degree program
11	Universität Regensburg	BAYERN-5 (BY)	
12	Universität Hamburg	HAMBURG-1 (HH)	
	Technische Universität Hamburg	HAMBURG-2 (HH)	hochschulkompass.de
	Hafencity Universität, Hamburg	HAMBURG-3 (HH)	hochschulkompass.de
13	Universität Frankfurt am Main	HESSEN-1 (HE)	
14	Universität Gießen	HESSEN-2 (HE)	
15	Technische Universität Darmstadt	HESSEN-3 (HE)	
16	Universität Marburg	HESSEN-4 (HE)	

Table 4. *Cont.*

No.	University (GENESIS Name)	Federal State with Rank	Exclusion Criterion
17	U Kassel in Kassel (ab 1999 ohne Kunsthochschule)	HESSEN-5 (HE)	
18	Universität Rostock	MECKLENBBURG-VORPOMMERN-1 (MV)	
19	Universität Greifswald	MECKLENBBURG-VORPOMMERN-2 (MV)	
20	Universität Hannover	NIEDERSACHSEN-1 (NI)	
21	Universität Göttingen	NIEDERSACHSEN-2 (NI)	
22	Technische Universität Braunschweig	NIEDERSACHSEN-3 (NI)	
23	Universität Oldenburg	NIEDERSACHSEN-4 (NI)	
24	Universität Osnabrück	NIEDERSACHSEN-5 (NI)	
25	Universität Köln	NORDRHEIN-WESTFALEN-1 (NRW)	
26	Technische Hochschule Aachen (U)	NORDRHEIN-WESTFALEN-2 (NRW)	
27	Universität Münster	NORDRHEIN-WESTFALEN-3 (NRW)	
28	Universität Bochum	NORDRHEIN-WESTFALEN-4 (NRW)	
29	Universität Bonn	NORDRHEIN-WESTFALEN-5 (NRW)	
30	Universität Mainz in Mainz	RHEINLAND-PFALZ-1 (RP)	
31	Technische Universität Kaiserslautern	RHEINLAND-PFALZ-2 (RP)	
	Universität Trier	RHEINLAND-PFALZ-3 (RP)	hochschulkompass.de
32	Universität Koblenz-Landau in Landau	RHEINLAND-PFALZ-4 (RP)	
33	Universität des Saarlandes Saarbrücken	SAARLAND-1 (SAR)	
34	Universität Leipzig	SACHSEN-1 (SA)	
35	Technische Universität Dresden in Dresden	SACHSEN-2 (SA)	
	Technische Universität Chemnitz	SACHSEN-3 (SA)	hochschulkompass.de
	Technische Universität Bergakademie Freiberg	SACHSEN-4 (SA)	hochschulkompass.de
36	Universität Halle in Halle	SACHSEN-ANHALT-1 (SAH)	
37	Universität Magdeburg	SACHSEN-ANHALT-2 (SAH)	
38	Universität Kiel	SCHLESWIG-HOLSTEIN -1 (SH)	
	EUF Europa-Universität Flensburg	SCHLESWIG-HOLSTEIN -2 (SH)	hochschulkompass.de
	Universität zu Lübeck	SCHLESWIG-HOLSTEIN-3 (SH)	hochschulkompass.de
39	Universität Jena	THÜRINGEN-1 (TH)	
	Universität Erfurt	THÜRINGEN-2 (TH)	hochschulkompass.de
	Technische Universität Ilmenau	THÜRINGEN-3 (TH)	hochschulkompass.de
	Bauhaus-Universität Weimar	THÜRINGEN-4 (TH)	hochschulkompass.de

In Table 4, we assign unique numbers only to those universities that were included in the sample for analysis.

The actual number of module handbooks per university can vary slightly and depends on the degree awarded. In Germany, it is common practice to divide the teacher training program into a bachelor's degree program with a consecutive master's degree program. Even if students can obtain a master's degree at another university after completing a bachelor's degree, these two courses are coordinated at the respective universities. The master's degree takes up specific competencies and content from the bachelor's degree and expands them. The consequence is that the intended professionalization is only achieved in certain sub-competencies if a student attends both courses at the same university. Therefore, we document in our analysis whether computational literacy is described in the module manual of the bachelor's or master's degree program. Most degrees are Bachelor of Education and Master of Education. In some states, it is also common to study a Bachelor of Science or Bachelor of Arts followed by a Master of Education. In this case, the Bachelor of Science or Arts degree is extended by modules specific to the teaching profession, with subject-specific modules being deleted to compensate. In a few federal states, there are instead coherent teacher training courses. A bachelor's degree usually comprises six terms, and a master's degree is four terms, so the total duration of study in teacher training programs for K12 teachers is usually between nine and ten terms.

In Table A1 (see Appendix A), we document the sources of the module handbooks. We used the latest version of a module book available in the winter term 2022/2023. We also considered versions that will only be valid in the coming terms insofar as they have already been legally passed. As a matter of principle, we did not examine draft versions. The numbers in Table A1 reference the universities as listed in Table 4. The validity of the links was last checked on 10 May 2023.

Limiting is the fact that the presentation and depth of description in the module handbooks of different universities can vary greatly, and sometimes, only a few key points are listed in tabular form. For our approach, however, only these module handbooks come into question since they are (mostly) freely accessible as course descriptions at almost all universities and strive for comparability, at least to some extent. In addition, in contrast to descriptions from the term course catalogs, they are not directly dependent on just one teacher and are valid beyond single terms. Another aspect limiting usability is that due to the practiced freedom of teaching in Germany, these descriptions, unfortunately, do not necessarily have to be implemented by the teachers, even though this would be desirable. This can make our findings less representative and objective overall. These shortcomings are discussed in more detail in Section 8. The lack of alternatives causes the module handbooks to be the best option for our research. Because of the relatively high number of universities distributed across the country, we can still make statements and show the current situation.

6. Results

When evaluating the module handbooks, we distinguished between the bachelor's and master's degree programs. However, we treat both partial degree programs as a single unit during the teacher training phase since a consecutive master's degree program is usually required for a qualifying teacher training program. This is because universities typically assume that students will complete both programs in succession at the same university as part of their study program design. In addition, we support comparability with study programs that have not introduced a corresponding division of teacher training into two parts. Suppose that we nevertheless distinguish between bachelor's and master's degrees in concrete individual considerations. In that case, we explicitly point this out, e.g., whether the respective competencies are already intended to play a role at the beginning or toward the end of the study program. To facilitate the overview of the relevant abbreviations used, we summarize all relevant abbreviations in Table 5, including brief explanations.

We structure the presentation of our results based on our research questions, with concrete quantitative results given in Table 6.

In Table 6, except for the "W/WP" column, only mandatory courses and events are included. Therefore, an "x" in these columns also means that aspects of computational literacy were found in there. An "x" in the "W/WP" column means that other relevant courses are described in the module manual. We refrain from further differentiation here due to the explanation in Section 4. In the remaining columns, an "x" stands for the fact that the respective statement is (mostly) covered by compulsory courses, and a "W" accordingly indicates that the competencies can be completed mainly in the elective area. Accordingly, a “B” in the second-to-last column means that the focus on computational literacy is more in the bachelor’s phase, and an “M” means that it is emphasized in the master’s phase. If there is no indication here, no focus was identified. The “Code/CAS” (computer algebra system) column indicates whether competencies or knowledge in this area are specifically trained or required in modules.

RQ1: In which physics content or modules does computational literacy play a role?

Table 5. List of used abbreviations in the evaluations.

Abbreviation	**Explanation**
C-Theo	Physics content, theoretical physics. Content-knowledge courses focusing on theoretical physics.
C-Exp	Physics content, experimental physics. Content-knowledge courses focusing on experimental physics.
C-Lab-Beg.	Physics content, beginners’ physics lab. Content-knowledge laboratory courses for beginners.
C-Lab-Adv.	Physics content, advanced physics lab. Content-knowledge laboratory courses for advanced students.
CP-C	Physics didactic content. Pedagogical-content courses focusing on didactics of the subject.
CP-Lab	Physics didactics with physics lab. Pedagogical-content courses focusing on didactics of experiments.
Focus DMA	Focus on digital data acquisition/measurement. Courses with a strong emphasis on digital measurement acquisition.
SMC	Special mandatory courses. Courses from specialization areas or in specific research groups.
Code/CAS	Focus on (scientific) coding or use of computer algebra systems.
AUT	Overall focus on Automation (see Table 2).
DIG	Overall focus on Digitization (see Table 2).
CSS	Overall focus on Information Technology Systems (see Table 2).
Focus PKN	Focus on product knowledge. Teachers remain in the user’s role of specific digital tools.
CS explicitly	Computer science elements or competencies explicitly mentioned.
W/WP	Elective (W, German: “Wahlbereich”) or compulsory elective (WP, German: “Wahlpflichtbereich”) courses.
Focus B/M	Focus on bachelor’s (B) or master’s (M) phase.

From a physics perspective, Figure 3 shows that in 56.4% of the study programs we analyzed, computational literacy is mainly taught in digital measurement or data acquisition. Students only perceive it as a basic skill within this limited context. In 56.4% of the cases, an emphasis on “computer-supported” or “computer-aided” apparatuses or measured value recordings is found. From the perspective of independent experimentation, fundamental aspects are mostly built up in the beginner or basic practical courses (25.6%) and then are either deepened with a focus on school experiments in physics education courses (35.9%) or focused on more complex setups in subject areas of modern physics (20.5%). The objective in this scenario is to reinforce computational literacy within subject-specific advanced practical courses.

Table 6. Results.

	RQ1									**RQ2**				**RQ3**			
No.	**C-Theo**	**C-Exp**	**C-Lab-Beg.**	**C-Lab-Adv.**	**CP-C**	**CP-Lab**	**Focus DMA**	**SMC**	**Code/CAS**	**AUT**	**DIG**	**CSS**	**Focus PKN**	**CS explicitly**	**W/WP**	**Focus B/M**	**Federal State**
1			x			x	x					x					BE
2				x			x					x	x		x	M	BE
3		x	x				x					x				B	BB
4	x					x						x	x		x		BR
5						x						x				B	BW
6			x			x	x						x			M	BW
7				x	x		x			W		x			x		BW
8					x		x			W			x	W	x		BW
9				x		x	x				x	x		x			BW
10				x	x	x			x	x		x	x	x		-	BY
11					x			W	W	W	W	x		W	x	-	BY
12						x	x					x	x			B	HH
13						x	x					x	x			-	HE
14						x	x					x	x			-	HE
15					x	x					x					-	HE
16					x		x					x	x			-	HE
17																-	HE
18								x	x	x		x		x	x	-	MV
19				x				x	x	x	x	x	x	x		-	MV
20	x			x		x			x	x	x	x	x	x			NI
21			x		x	x			x	x		x		x		B	NI
22								x	x	x				x			NI
23							x					x	x		x	B	NI
24					x												NI
25					x	x				x		x		x		M	NRW
26		x					x						x				NRW
27					x			x			x	x		x			NRW
28		x			x				x	x	x	x		x	x		NRW
29										W	W	W			x		NRW
30			x		x		x	x				x	x			B	RP
31			x		x		x					x	x		x		RP
32			x		x		x	x				x	x				RP
33			x	x					x	W	x		x	x	x	-	SAR
34							x									-	SA
35					x		x						x			-	SA
36		x			x		x				W		x		x	-	SAH
37			x		x		x	x			x	x	x	W	x		SAH
38								x			x					B	SH
39				x			x			W		x	x			-	TH

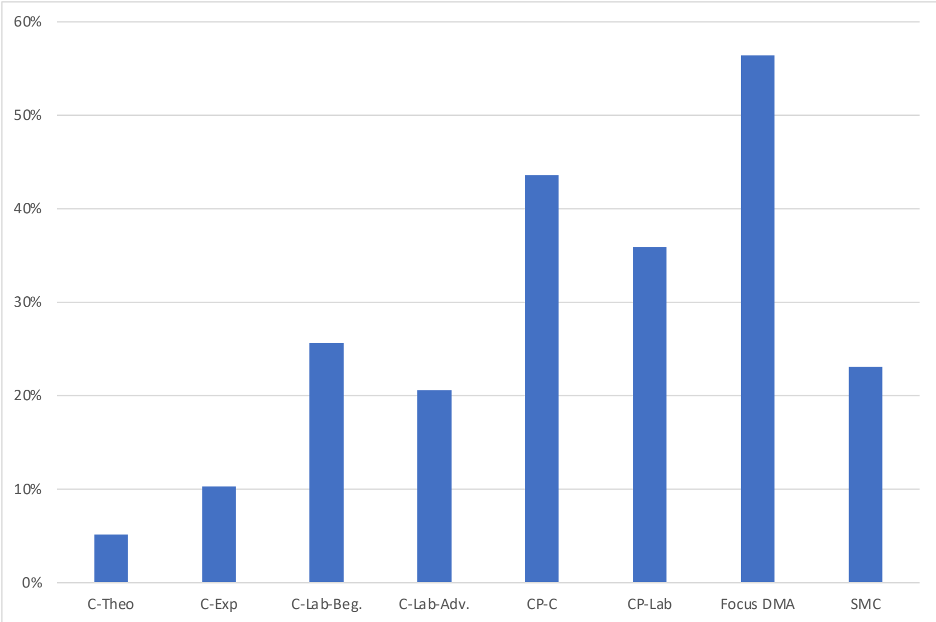

Figure 3. RQ1: In which physics content or modules does computational literacy play a role?

Overall, we argue that most courses (82.1%) have computational literacy in the context of student laboratory courses (see Figure 4). In addition, Figure 4 shows three other important aspects: First, the proportions of computational literacy in subject didactics without laboratories (CP-noLab, 43.6%) are roughly comparable to those in subject laboratories without subject-didactic aspects (Lab-noCP, 46.2%). From this, we can deduce that equivalent emphases are set and thus that training is not too one-sided at the expense of the other. Secondly, these ratios are balanced within the sub-disciplines (CP-noLab/CP and Lab-noCP/Lab). Thirdly, the contribution from all subject courses without laboratories (CK-noLab, 15.4%) is underrepresented, limiting subject expertise. Thus, we conclude from Figure 4 that student teachers acquire computational literacy primarily through teaching-specific courses. If we exclude the subject-specific internships, it becomes even clearer that the acquisition of competencies related to computational literacy is much more underrepresented in subject-specific courses than contemporary physics education allows. Furthermore, we can deduce from many module descriptions that computational literacy plays a role in the evaluation, analysis, and presentation of digital measurement results, whereby the descriptions already refer explicitly to special hardware and especially software in some cases.

Some study programs (23.1%) describe modules in which measurement technology or "computer physics" is the main content. In some cases, practical courses like "electronics laboratory" or numerical exercises are also included in teaching computational literacy. These courses focus on skills related to hardware components or digital circuitry. They are usually elective and not mandatory. However, it is worth noting that approximately half of these courses do not include any specific mention of programming or computer algebra skills.

In a clear minority of the courses considered, computational literacy appears in the descriptions of purely content-knowledge courses (15.4%, excluding laboratory courses; see Figure 4). On the other hand, these courses explicitly address mathematical knowledge, physical content, and competencies. This is especially true for the canonical lecture series on experimental and theoretical physics, which are comparably present in all study programs. This makes the small part mentioned above even more significant. On the other hand, there is a stronger tendency to take up aspects of computer literacy in special lectures from the elective area (33.3%). Then, for example, it is about solving physical problems with the help of the computer through concrete implementations of algorithms or programming and the visualization of numerical methods. Here, terms like "scientific computing" or "computer physics" appear in the module handbooks.

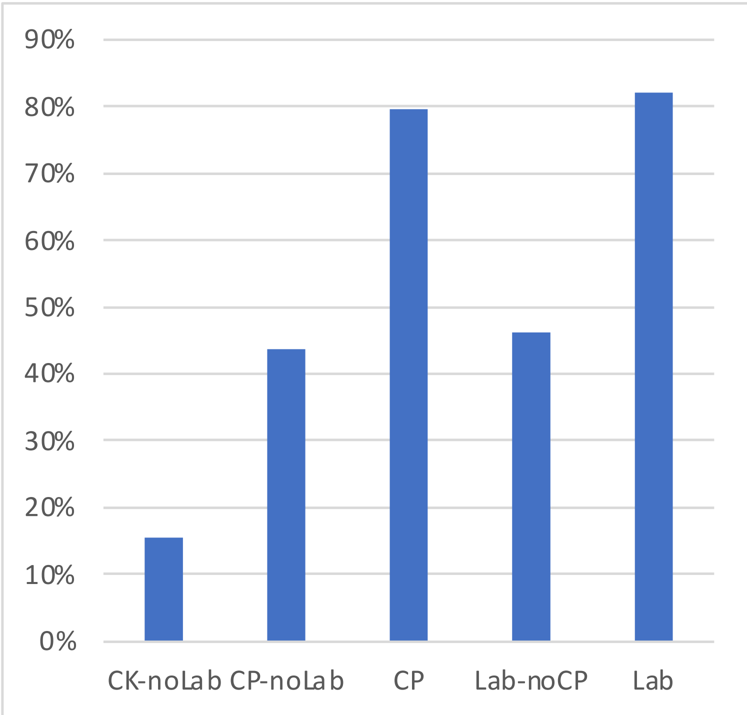

Figure 4. Cumulative representation of the course distribution. CK-noLab: CK courses excluding any laboratory courses; CP-noLab: CP courses excluding any laboratory courses regarding (experimental) laboratory didactics; CP: CP courses including all laboratory courses regarding (experimental) laboratory didactics; Lab-noCP: laboratory courses excluding any laboratory courses regarding (experimental) laboratory didactics; Lab: all laboratory courses including laboratory courses regarding (experimental) laboratory didactics.

Depending on the focus of a module, knowledge of a higher programming language or a computer algebra system is required (23.1%). However, students are supposed to acquire these integratively in most courses and do not have to acquire them as a prerequisite beforehand. Only in the laboratory courses and a few special lectures of the elective area are the programming skills taught ahead of the courses (e.g., basic or beginner laboratory courses) or subsequently required in advanced laboratory courses or courses in isolated study programs.

RQ2: Which computer science competencies or knowledge can be identified or derived?

The identified computational literacy skills are distributed among the previously mentioned basic concepts as follows: automation 35.9% (without elective courses, only 20.5%), digitization 30.8% (without elective courses, only 23.1%), and computer science systems 71.8% (without elective course, 69.2%). We illustrate this distribution in Figure 5. With the view of the partial aspects, we derive that these often occur in combination. Without consideration of the computer science systems, the two other basic concepts appear only in four cases isolated; otherwise, at least two basic concepts are represented (see Table 6). These courses primarily focus on how computer science systems can digitally record, process, and store measured values, considering issues such as finite computer accuracy. In physics, these aspects are closely related to technology, such as sensors or digital circuits and gates.

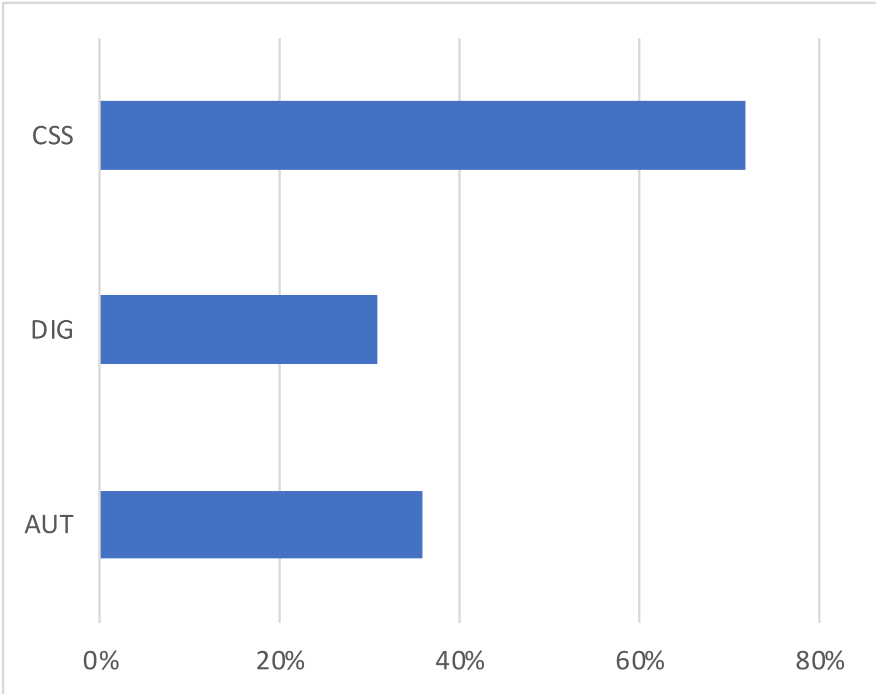

Figure 5. Distribution of computational literacy skills according to basic computer science concept topics.

Additionally, aspects of Boolean algebra play a role in the use and combination of circuits and components. Based on the descriptions in the module handbooks, algorithms as a sub-aspect of automation are linked to knowledge of a programming language in 25.6% of the cases. In most cases, it remains an open question as to how deeply these competencies are developed. A few modules here also deal with concepts and modeling. Thus, it cannot be excluded based on module descriptions that programming is taught as assembling (given) code snippets.

The concept of computer science systems has a special role here. Very often, only the aspect of data acquisition systems is described for data-processing systems in the module handbooks (see also results of RQ1). However, computer science systems and the related computational literacy content are much broader. Thus, if you wish to classify the relevance of this basic concept in the courses of study, you must note that the proportion of computer science systems without a predominant focus on the handling of measured values is only 15.4% (only 12.8% without electives).

RQ3: Are computer science competencies or knowledge described implicitly or explicitly?

By distinguishing between implicit and explicit descriptions of computational literacy and computer science content, we made three main observations, also shown in Figure 6. First, computational literacy is only implicitly included in most module handbooks (64.1%). Even then, we can mostly derive them from explicitly physically described content such as "computer-based experimentation with a measurement acquisition system" or "basics of data processing" only. This is because the reference to "modern measurement technology" or "digital media" does not necessarily stand for the teaching of relevant computational literacy (see Section 4). Second, a comparison between explicitly described mathematical and computer science competencies and contents shows that module descriptions can be very specific about the mathematical skills and knowledge necessary for a successful physics study. However, much remains intentionally unspecified concerning computational literacy education, and no binding requirements are given since they are not explicitly specified. Third, some module handbooks show impressively how computational literacy skills or even computer science competencies can be explicitly included from the physics perspective for a successful teacher education program. However, these courses often reside in the elective area (e.g., university 11, 19, 25, 33, or 37). Nevertheless, such explicit descriptions are found in only 35.9% of the module handbooks considered.

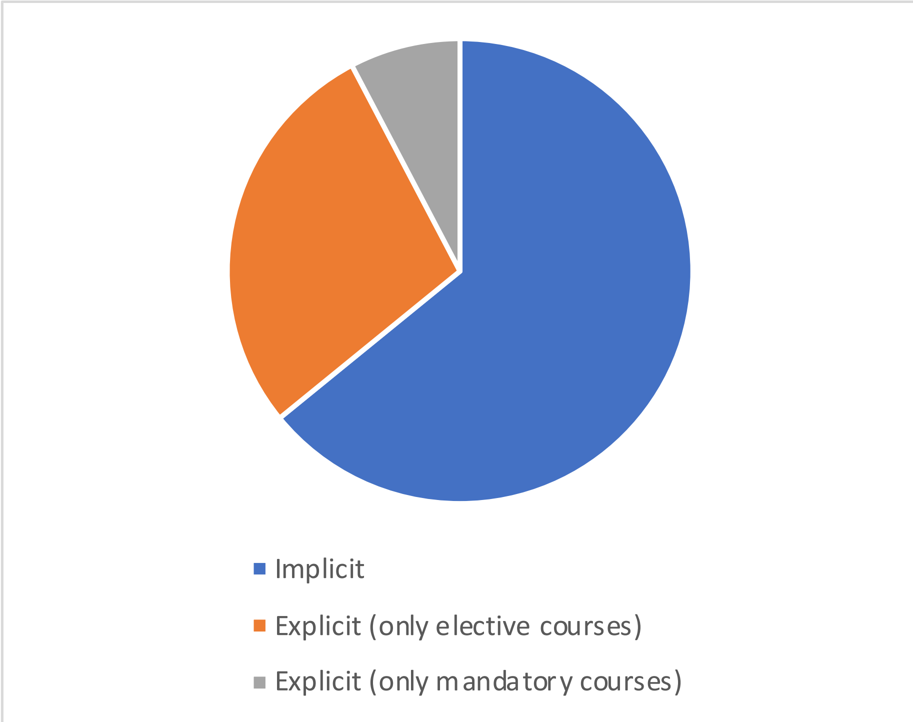

Figure 6. RQ3: Are computer science competencies or knowledge described implicitly or explicitly?

7. Discussion and Implications

As already mentioned, to our knowledge, there are currently no comparable scientific studies of science teacher education programs with these questions in the international environment. However, publications in other disciplines can be found in the international literature, which systematically evaluates module handbooks as a data source to derive patterns and findings from a curricular perspective [54–59]. In this context, our research approach on the one side and our results and conclusions on the other are internationally connectable and comparably underpinned.

Based on our remarks on the fundamental importance of computational literacy in the 21st century in Section 1 and the theoretical meaning for science education in Section 2, our results show that computational literacy or computer science basic concepts and knowledge are included in the module handbooks of physics teacher training programs to some extent. However, upon closer analysis, we identify some significant shortcomings: the close connection to mandatory laboratory courses makes it difficult to see that most computational literacy skills are only covered in the module handbooks regarding digital measurement acquisition. Computer science systems for measurement acquisition in physics may have an important role. However, the isolated focus on this application-oriented sub-branch of computer science systems does not ensure the sustainable and broad competence development of computational literacy as required by the extended DPACK model. In our opinion, teaching competencies predominantly only with that narrow orientation does not go far enough. For example, from a computer science perspective, the description of a digital measurement process is incomplete if, in addition to the measurement system, aspects of coding and file formats are not included as partial aspects of digitization. However, according to our results, the basic concepts of digitization and computer science systems occur together in only 15.4% of the cases considered (excluding elective courses). The situation is similar for the pairing automation and computer science systems. Here, it is 18.0%, with intersections with the first pairing. In particular, according to our evaluation, the algorithms' sub-aspect (implementation) has too low a prevalence, with only 20.5% in the compulsory area. When advanced physical problems or experiments are taught in the "modern physics" field, a more complex algorithm is necessary to incorporate computer systems or computer science concepts. This can be described as a modern problem-solving technique. Our first research question leads to two conclusions: First, computational literacy can be sufficient when at least two basic concepts are adequately present. Ideally, according to our analysis, all three concepts are mandatory in the module handbook, which is true for only 7.7% of the module handbooks (see Table 6). Second, partial aspects of computational literacy must be included in the module descriptions of the canonical lectures, especially in the beginner lectures on theoretical and experimental physics. Otherwise, if the courses are designed accordingly, it cannot be ruled out that students associate computational literacy in physics only with experimenting with the help of digital systems. For example, robots as educational objects can enable the interlocking of many aspects, such as problem-solving skills, critical thinking, and collaboration—especially in STEM subjects or competitions in a wider context [60].

The results of the second and third research questions (RQ2 and RQ3) also show that the implicit inclusion of computational literacy is fundamentally problematic. On the one hand, we can infer a lack of commitment to the implementation, so computational literacy is only taught superficially. However, sustainable and explicit integration is necessary. This becomes especially clear when the module manuals speak, for example, of "knowledge in the use of software". It becomes even more obvious when product names such as "Matlab" or "Origin" appear in concrete descriptions of competencies. In this way, students receive primarily product knowledge that cannot be used sustainably in professional practice, especially if such software tools change or are unavailable at schools. Here, we strongly advocate that by explicitly formulating computational literacy skills and computer science knowledge, sustainable concept knowledge is focused on instead. Suppose that skills are only implicitly included in module handbooks. In that case, there is usually a greater risk that product knowledge and concept knowledge will be disproportionately biased in favor of product knowledge (of the teacher).

On the other hand, a comparison with the formulated mathematical competencies shows the direction in which things can evolve. For example, explicit formulations, especially in the basic lectures on experimental and theoretical physics, show the importance that a subsidiary discipline can have in module handbooks. If we assume, based on our considerations in Section 2, that computer science is also necessary as a subsidiary discipline like mathematics in order to understand physics and to be able to teach it later in school, then module handbooks must include more and especially more explicit formulations on computational literacy. This is because we did not find any module handbook that uses only implicit formulations when describing the required mathematical competencies. For example, suppose that we must explicitly use application skills and knowledge of mathematical integral theorems in a theoretical lecture to teach Maxwell's equations. In that case, that also needs to be explicitly described in the module handbooks. To ensure that students have the necessary prerequisites for computational literacy, it is important to approach it in the same way as mathematical literacy (cf. [61]). This means identifying and communicating the requirements for computational literacy in study manuals, which we believe is not covered by most of the handbooks that we examined. While some may argue that computational literacy is included in demands for mastering relevant methods and competencies in physics, we disagree. Computational literacy does not arise solely from physics methods and must be addressed as a distinct skill set. It arises from the discipline of computer science and the increasing digitality of research and teaching in almost all areas of society in which physics is also relevant, as we explain adequately in Section 2. Second, the statement would imply that what is relevant in the context of computational literacy is already known. However, our evaluation of the module handbooks has impressively shown that (at least on the formal level of the module handbooks) it is still completely unclear and unspecified which computational literacy skills and knowledge are to be mastered by students. Third, there is a consensus that it is not sufficient for module handbooks to anchor mathematical literacy exclusively with unspecified descriptions such as "students master the relevant methods and competencies of mathematics".

From the education perspective, the above pairing of Maxwell's equations and integral theorems shows another gap, which we can recognize in the module manuals considered. Suppose that we disregard the broad topic of acquiring and processing measured values. In that case, the descriptions lack concrete physical content that could be directly related to computational literacy. There is an attempt to compensate for this lack of content level by using collective terms like "computer physics" or focusing on process-related competencies like "solving physical problems with the computer". Comparable course offerings, e.g., focusing on physical computing, can also be found internationally [62]. This leads to an increased arbitrariness in the implementation of such content depending on the preferences of the lecturers. However, especially concerning mathematical literacy in undergraduate physics courses, we deduce that no effective and sustainable integration of computational literacy is possible without comparable content pairings. It follows that, as a starting point, descriptions in module handbooks must place concrete content, e.g., from mechanics or quantum mechanics, directly related to computational literacy or computer science knowledge. Related to the module handbooks under consideration, the descriptions of mathematical literacy in the canonical lectures can serve as a blueprint. Complementarily, descriptions of a few special physics courses also show how certain basic concepts of computer science already occur in a pairing with physics content.

The results also show a difference between subject-specific and physics education courses. Based on the sample of physics education courses that we considered (as discussed in Section 6), there is a noticeable emphasis on computational literacy. As a result, we can make two primary conclusions: First, computer science competencies and knowledge, in a broader sense, may often be associated with media didactics or the integration of digital media such as computers in physics education. Thus, in contrast to purely subject-specific events, subject-specific didactic events also have an effective connection point independent of the physics content. This also highlights the importance of computational literacy for education (in physics) since it is already more strongly linked to teaching–learning processes and their didactic preparation through social processes. Positively formulated, we can thus organically further increase this significance by more explicit competence formulations in the module handbooks. On the other hand, this may imply that prospective teachers are less able to retrieve knowledge and competencies in the classroom, which need to be linked to subject education from a scientific perspective. Thus, there is a particular danger that teachers will be able to acquire less computational literacy linked to subject matter science. This point is relevant because we only considered programs with the qualification goal of secondary school teaching, and this type of school focuses on preparing students for academic careers. For this reason, our results suggest a gap in module handbooks here as well. This could be further explored by looking at module handbooks for non-teaching programs. Therefore, more subject-specific computer science competencies and knowledge could already be integrated but without systematically benefiting most student teachers.

To conclude this section, we analyze our results from the point of view of the state affiliation of the universities. From a structural point of view, this aspect is interesting because both physics as a subject science and the need to integrate computational literacy into science education are independent of external conditions, such as the political requirements of public school education. However, implementing study programs, especially teacher training, is not true internationally outside of Germany. Due to the fundamentally federal structure in Germany regarding education, there are thus 16 different framework conditions that are supposed to pursue the same goal. This puts the validity of our analysis on a more differentiated data basis, which facilitates a transfer to different international framework conditions. This aspect is further supported by the fact that within a federal state, the educational path of a K12 teacher is structured as follows: First, students in (high) schools learn computational literacy according to the curriculum. Second, building on this foundation, university students are again committed to the curriculum from a teacher's role perspective. In this respect, the federal-state perspective is relevant to professional training. This is similarly structured in many other countries. Our results show certain peculiarities in the distribution if we exclude the ubiquitous focus on computer systems as a component of experimental education. We can derive the following striking statements: Basically, universities seem to exploit their leeway, the so-called freedom of teaching, to set independent accents. The module manuals of the study programs from the federal state with the abbreviation RP are the only counterexample: the descriptions of the module manuals of this state are strikingly similar in wording and structure. Here, concepts and contents seem to have been developed or adopted together. This is also evident in Table 6. Only in two states (MV, NI) are basic concepts of automation taught in most of the sample. If we add the elective area, this list is extended by three more states (BY, NRW, SAR, TH).

Thus, automation is underrepresented in 62.5% of the states. The basic concept of digitization is only present in two federal states with more than one university in the sample (SAH, NRW). From this unsystematic, unequal distribution of the basic concepts in the module handbooks, we deduce that, from a subject perspective, there should be more agreements and uniform standards and formulations so that the development of computational literacy becomes more independent of location. In conclusion, 64.1% of the (considered) study programs are divided into bachelor's and master's degrees. The majority (60.0%) of these programs distribute their offerings on computational literacy without an explicit focus on one of these phases, whereas 28.0% of the module handbooks bundle the offerings in the bachelor's program, and 12% include them in the master's program. Thus, we conclude that, overall, there is a relevant tendency to ensure that competencies are not underrepresented In the bachelor's degree. Since practical teacher training at internship schools focuses mostly on the master's phase, we conclude that prospective teachers should have already acquired knowledge on this aspect before they are assigned to teach in schools.

8. Limitations

In many educational landscapes, freedom of teaching applies, and universities have a broad scope for shaping the implementation of their courses of study. In addition, there are often political requirements because school education and, thus, teacher training are considered public tasks. We decided to use module handbooks as a starting point for these reasons. More specifically, the reason for choosing module handbooks as the focus of our analysis is twofold. Firstly, they provide a baseline for defining the content and competencies of a particular course of study within a university. Secondly, they have a consistent, standardized structure and content across most educational institutions, making comparing patterns and identifying differences easier. In particular, the explicit (subject-specific) bundling and description of competencies as a common basis are relevant here. In contrast, two major disadvantages are already clear in advance: First, module handbooks do not explain which competencies exactly are necessary and, above all, in what depth a lecturer deals with the described aspects in a specific course. However, our interest refers to the fundamental integration of computational literacy into teacher training in physics; this must be independent of the preferences of individual lecturers. In this respect, module handbooks at least passed through different committees and instances consisting of subject experts and responsible persons. Therefore, it can be concluded that selecting computational literacy skills based on module handbooks is more sustainable and meaningful, particularly for training subject-specific teachers. Module handbooks have the advantage of explicitly defining competencies for a specific course of study, allowing for a clear distinction between teaching and non-teaching courses, even in an international context. However, it should be noted that module handbooks only outline the intended competencies for each module and not the actual competencies acquired by students. Therefore, competence tests are necessary to measure the increase in competency. Currently, such competence tests do not exist, as research on the intersection of computer science in science education is still in its infancy, and this paper contributes to identifying relevant competencies.

After analyzing the module handbooks, we found that many study programs have included information technology systems as a sub-aspect. However, it is possible that this only updates the concept of experimentation with new devices and technical developments without necessarily incorporating computer science competencies and conceptual knowledge. Therefore, the module descriptions do not guarantee the inclusion of these competencies and knowledge. Therefore, there is a probability that the rate of 71.8% significantly exceeds the actual percentage of computational literacy. In order to classify this statement, in the analysis and discussion of the results, we have also determined a ratio that shows to what extent the description suggests that the focus is too much or predominantly on pure measurement or experimentation. This is the case in 77.3% of the courses with a compulsory course in computer science systems. That a limitation may be implied here by the source material is also evident from the fact that the other basic concepts of computational literacy are significantly less represented (see Section 6).

Finally, we want to point out that computational literacy alone is insufficient for student teachers to sustain the required competencies in STEM education. On the one hand, better framework conditions must also be created in the training structure, profession, and technical equipment. On the other hand, the significance and use of digital media in the sense of computers must also be constantly practiced, evaluated, and reflected upon. Especially regarding STEM subjects, a constant exchange among the teachers and with the training centers is useful to support this process dynamically. Therefore, while computational literacy is fundamentally a prerequisite, it is not sufficient for mastering the use of digitality in STEM subjects and education. However, these other aspects are not the subject of this paper.

9. Conclusions and Outlook

Contemporary science education in the 21st century requires computational literacy. For this reason, it must be integrated into the associated (preservice) teacher training courses. Hence, the essential ideas and skills of computer science should be adequately encoded in current module descriptions, which could be included as an attachment to the mandatory examination regulations. As we have presented, the first two statements are accepted as true. The extended DPACK model theoretically underpins them. However, our analysis of the current state of the module handbooks of physics (preservice) teacher training programs in the 16 states of Germany shows that there are still large gaps and, in some cases, differences that hamper or even prevent a sustainable and balanced anchoring of computational literacy. This raises the question of the future viability of this field of study and, thus, a cornerstone of teacher training. These fundamental challenges and preconditions also apply internationally to other countries since the digital transformation in science education also plays a decisive role there. So, the question is, What aspects of computational literacy will a teacher need in the future to be able to teach physics? At least in a broad outline, module handbooks should offer possible answers as a minimum consensus among teachers.

With our systematic analysis of the module handbooks, we identified the first significant results based on the current status of the study programs using our three research questions. We showed that computational literacy is mainly addressed in laboratory courses and relatively often in subject-didactic courses. However, the descriptions of the purely subject-oriented and basic lectures significantly lacked concrete computational literacy skills or knowledge. There is a small number of special courses addressing related topics. However, these are often only taken as electives, depending on the students' interests. Moreover, this isolated teaching lacks the aspects of universality and sustainability, both of which are essential for effective skill development. The results of the other two research questions show that relevant computer science competencies are only implicitly described in connection with physical contents in most of the considered study programs (64.1%).

This means that the actual proportion of computational literacy taught is anchored in too few binding and transparent forms.

Moreover, most module handbooks are limited to stipulating only one specific aspect of computational literacy as an isolated focus: the basic concept of computer science systems in the very narrow understanding as "computer-based" or "computer-assisted" measurement acquisition systems. Based on our discussion and conclusions, we suspect that in this way, computational literacy is either taken up too one-sidedly and superficially without technical anchoring or with only product knowledge about measurement tools, which inevitably becomes obsolete. This hinders the concept-oriented and sustainable integration that contemporary science (teacher) education demands. In addition, we see a further gap regarding the inclusion of important basic concepts such as automation and digitization from the perspective of computational literacy. While we have sufficiently explained the basic limitations that an analysis of module handbooks entails, we nevertheless believe that, based on our analysis, there are important trends and gaps for future development in the systematic integration of computational literacy into science teacher education.

Looking forward, we would like to propose two potential research objectives. First, it would be valuable to investigate the extent to which computational literacy is a necessary auxiliary science, much like mathematics, and should be explicitly included in the physics courses of future teachers. This investigation could examine how extensively computational literacy should be integrated into module manuals for physics courses. Second, this objective is related to the extent to which science teachers need explicit computer knowledge and skills while maintaining a strong connection to physics and subject language or content. Our research shows that there are still gaps in the identification and definition of relevant concepts and terms in the module manuals considered in all 16 states. Therefore, we conclude that, currently, the teacher education of physics teachers still insufficiently maps and integrates computational literacy. Matching these gaps with module handbooks that already take computational literacy into account in a more differentiated and explicit way can offer potential approaches to improve the (science) curricula. Theoretical models such as DPACK can also provide an important impetus. After all, the goal must be to further develop the existing curricula, which currently do not teach all of the necessary computational literacy skills for (science) teachers to meet the requirements of a digitized society.

Author Contributions: D.B.: conceptualization, methodology, investigation, data curation, formal analysis, writing—original draft, writing—review and editing, and visualization. J.H.: resources, writing—review and editing, supervision, and project administration. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Federal Ministry of Education and Research (project "edu4.0") in the framework of the joint "Qualitätsoffensive Lehrerbildung", grant number 01JA2011 and project "MINT-ProNeD", grant number 01JA23M02K. The APC was funded by the University of Konstanz.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: See Table A1.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Appendix A

Table A1. List of sources of the module handbooks.

No.	Study Degree	Source (URL Accessed on 10 October 2023)
1	B.Sc. M.Ed.	https://www.fu-berlin.de/service/zuvdocs/amtsblatt/2015/ab342015.pdf https://www.fu-berlin.de/service/zuvdocs/amtsblatt/2018/ab272018.pdf
2	B.Sc. M.Ed.	https://gremien.hu-berlin.de/de/amb/2014/58/58_2014_AMB_KombiBA_Physik_DRUCK.pdf https://gremien.hu-berlin.de/de/amb/2015/63/63_2015_MA_Physik_Gym_2015-06-17-Boc_PSE_7.8.15_DRUCK.pdf
3	B.Ed. M.Ed.	https://puls.uni-potsdam.de/qisserver/rds?state=verpublish&status=transform&vmfile=no&moduleCall=ModulkatalogAnzeigen&publishConfFile=modulkatalog&publishSubDir=up/modulkatalog&modulkatalog.mk_id=123&xslobject=pdf1 https://puls.uni-potsdam.de/qisserver/rds?state=verpublish&status=transform&vmfile=no&moduleCall=ModulkatalogAnzeigen&publishConfFile=modulkatalog&publishSubDir=up/modulkatalog&modulkatalog.mk_id=118&xslobject=pdf1
4	B.Sc. M.Ed.	https://www.uni-bremen.de/fileadmin/user_upload/fachbereiche/fb1/fb1/FB1/Pruefungsamt/Modulhandbuecher/Modulhandbuch_BSc_Physik_Vollfach_BPO_2020_v1_2_13102021.pdf and https://www.uni-bremen.de/fileadmin/user_upload/fachbereiche/fb1/fb1/FB1/Pruefungsamt/Modulhandbuecher/Modulhandbuch_Physik_ZF_BPO_2020_v1_1.pdf https://www.uni-bremen.de/fileadmin/studydata/DBS/Dokumente/Studienverlaufsplaene/FB1/Modulhandbuch_Physik_Master_of_Education.pdf
5	B.Ed. M.Ed.	https://uni-tuebingen.de/securedl/sdl-eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpYXQiOjE2ODM2NjgzOTQsImV4cCI6MTY4Mzc1ODM5NCwidXNlciI6MCwiZ3JvdXBzIjpbMCwtMV0sImZpbGUiOiJmaWxlYWRtaW5cL1VuaV9UdWViaW5nZW5cL0Zha3VsdGFldGVuXC9NYXROYXRcL0ZhY2hiZXJlaWNoZVwvUGh5c2lrXC9TdHVkaXVtXC9CRWQtTUVkX1BoeXNpa1wvTUhCX1BoeXNpa19fU3RhbmRfMjAyMy0wMi0wMl8ucGRmIiwicGFnZSI6MTc4NzEzfQ.0q5-R7X9wnRF7saHRtn-FBVtUYXCZplCEqXsRIbMszs/MHB_Physik__Stand_2023-02-02_.pdf
6	B.Ed. M.Ed.	https://www.physik.uni-heidelberg.de/c/image/f/studium/bachelor/pdf/BSc-Module-2019_1.pdf https://backend.uni-heidelberg.de/de/dokumente/modulhandbuch-physik-med-teilstudiengang-2017/download
7	B.Sc. M.Ed.	https://www.physik.uni-freiburg.de/bilderunddateien/dateien/Poly_Modulhandbuch_01.05.23.pdf https://www.physik.uni-freiburg.de/bilderunddateien/dateien/MEd_Modulhandbuch_01.05.23.pdf
8	B.A. M.Ed.	https://www.student.uni-stuttgart.de/studiengang/Physik-B.A-00001.-Lehramt/?page=studienaufbau#studienaufbau-1-1-child https://campus.uni-stuttgart.de/cusonline/wbModhbReport.downloadPublicMHBVersion?pOrgNr=3&pStpStpNr=946
9	B.Ed. M.Ed.	https://www.physik.kit.edu/downloads/Lehramt_Dokumente/MHB_BA_22_23.pdf https://www.physik.kit.edu/downloads/Lehramt_Dokumente/MHB_MA_22_23.pdf
10	LA, Staatsexamen	https://www2.uni-wuerzburg.de/mhb/MHB1-de-L5-128-H-2020.pdf
11	LA, Staatsexamen	https://www.uni-regensburg.de/physik/fakultaet/studium/modulkataloge/la-gym/index.html
12	LASek	https://www.physik.uni-hamburg.de/dokumente/studiengaenge/modulhandbuchlamaster.pdf
13	Lehramt L3	https://www.uni-frankfurt.de/73469527/Lehramt_Physik_L3_2018_08_23.pdf
14	Lehramt L3	https://www.uni-giessen.de/de/mug/7/pdf/7_80/7_83/Anlage2/SVP/physik/7_83_00_18ae_SVP_Phys

Table A1. *Cont.*

No.	Study Degree	Source (URL Accessed on 10 October 2023)
15	Lehramt Gmyn.	https://www.zfl.tu-darmstadt.de/media/zfl/studium/lag/studienordnungen_1/physik/Physik_LaG_Modulhandbuch_10-2017_SB_2017-II.pdf
16	Lehramt Gymn.	https://www.uni-marburg.de/de/universitaet/administration/recht/studprueo/03-lehramt/stpo-l3-20182_lesefassung_1-aenderung_korr.pdf
17	Lehramt Gymn.	https://www.uni-kassel.de/uni/index.php?eID=dumpFile&t=f&f=1974&token=44a5de1064284cecfc8afb8857b1cd7956d935e0
18	Lehramt Gymn.	https://www.uni-rostock.de/storages/uni-rostock/UniHome/Gremien/Rechtsgrundlagen/Amtliche_Bekanntmachungen/2022/NR_38_2022.pdf and Online-Directory
19	Lehramt Gymn.	https://www.uni-greifswald.de/storages/uni-greifswald/2_Studium/2.4_Rund_um_die_Pruefungen/2.4.1_Pruefungs_und_Studienordnungen/Lehramt_modularisiert/Physik/PSO_LAG_Physik_2020.pdf
20	B.Sc. with M.Ed.	https://www.maphy.uni-hannover.de/fileadmin/maphy/Studium/Studierende/Modulkataloge/Modulkataloge_2022/Modulkatalog_Physik_LA_20221116.pdf
21	Two-Subject-B. M.Ed.	https://www.uni-goettingen.de/de/document/download/ddac530364106e8b475fd9f3f48a5739.pdf/Physik_AM2022.pdf https://www.uni-goettingen.de/de/document/download/1ef802d9b8b09acf2fd0be5281f149ba.pdf/ModulVZ_MA_of%20Education_2022_2.pdf
22	Two-Subject-B. M.Ed.	https://www.tu-braunschweig.de/fileadmin/Redaktionsgruppen/Fakultaeten/FK6/Ordnungen/Modulhandbuecher/2-F-BA_Modulhandbuch_2022-10-07_113336.pdf https://www.tu-braunschweig.de/fileadmin/Redaktionsgruppen/Fakultaeten/FK6/Ordnungen/Modulhandbuecher/Lehramt_GYM_Modulhandbuch_2022-10-07_114249.pdf
23	Two-Subject-B. M.Ed.	https://uol.de/f/5/inst/physik/PDF/Modulhandbuecher/Modulhandbuch_Zwei_Faecher_Bachelor_Master_of_Education_Physik.pdf
24	Two-Subject-B. M.Ed.	https://www.uni-osnabrueck.de/fileadmin/documents/public/ordnungen/PO-Master-Physik_2021-09.pdf
25	B.A. M.Ed.	https://physik.uni-koeln.de/fileadmin/Downloads/modulhandbuch/lehramt/Modulhandbuch_BA_GG_21.10.20.pdf https://zfl.uni-koeln.de/sites/zfl/ZfL-Navi/Modulhandbuecher/MNF/Master/MHB-M_Physik_GymGe.pdf
26	B. Lehramt M. Lehramt	https://online.rwth-aachen.de/RWTHonline/pl/ui/$ctx;design=ca2;header=max;lang=de/wbModhbReport.downloadPublicMHBVersion?pOrgNr=1&pStpStpNr=401&pDocNr=9328698 https://online.rwth-aachen.de/RWTHonline/pl/ui/$ctx;design=ca2;header=max;lang=de/wbModhbReport.downloadPublicMHBVersion?pOrgNr=1&pStpStpNr=430&pDocNr=8268426
27	Two-Subject-B. M.Ed.	https://www.uni-muenster.de/imperia/md/content/fachbereich_physik/studienordnungen/2023/po_zfb-physik_lesefassung_ws2023-24.pdf https://www.uni-muenster.de/imperia/md/content/fachbereich_physik/studienordnungen/2023/po_med-gymge-physik_lesefassung-ws2023-24.pdf
28	Two-Subject-B. M.Ed.	https://www.physik.ruhr-uni-bochum.de/wp-content/uploads/dateien/studium/modulhandbuecher-2-fach-bachelor/physik-modulhb-2-fach-bachelor-ws-22_23_stand01092022.pdf https://www.physik.ruhr-uni-bochum.de/wp-content/uploads/dateien/studium/modulhandbuecher-m.ed/physik-modulhb-m.ed_.-po-2020-ws-22_23_stand01092022.pdf
29	Lehramt B. M.Ed.	https://www.physik-astro.uni-bonn.de/de/studium/medienordner-studium-1/medienordner-studiengaenge/medienordner-lehramt/mh_bla-physik-2022.pdf https://www.physik-astro.uni-bonn.de/de/studium/medienordner-studium-1/medienordner-studiengaenge/medienordner-lehramt/download-msc-lehramt/mh_med-physik-2022.pdf

Table A1. *Cont.*

No.	Study Degree	Source (URL Accessed on 10 October 2023)
30	B.Ed. M.Ed.	https://www.blogs.uni-mainz.de/fb08-studium/files/2015/04/Mhb_BEd_Physik_2013.pdf https://www.studium.fb08.uni-mainz.de/files/2019/11/Modulhandbuch_MEd_Physik_Revision2019.pdf
31	B.Ed. M.Ed.	https://physik.rptu.de/fileadmin/uni_home/Modulhandb%C3%BCcher_und_Studienanleitung/2023/Modulhandbuch_Lehramt_Physik.pdf
32	B.Ed. M.Ed.	https://www.uni-koblenz.de/de/mathematik-naturwissenschaften/studium/infos/modulhandbuecher/emhb-phy-ba-ed/ https://www.uni-koblenz.de/de/mathematik-naturwissenschaften/studium/infos/modulhandbuecher/emhb-phy-ma-ed-gym/
33	Lehramt Sek1+2	https://www.uni-saarland.de/fileadmin/upload/fachrichtung/physik/Studium/Physik/Lehramt/Physik_LA-Modulhandbuch_20210818.pdf
34	Lehramt	https://db.uni-leipzig.de/bekanntmachung/dokudownload.php?dok_id=5383
35	Lehramt	https://www.verw.tu-dresden.de/amtbek/PDF-Dateien/2018-12/12_00soLAGP09062018.pdf
36	Lehramt	https://mos.uni-halle.de/Download/Aktuell/GB/MH_12446_aktuell.pdf
37	B.Sc. M.Ed.	https://www.math.ovgu.de/math_media/ordnungen/modulkatalog/BacLAallgSchulen2022_version1_9_final-p-17696.pdf https://www.bekanntmachungen.ovgu.de/media/Modulhandb%C3%BCcher/Master+_+Studieng%C3%A4nge/Lehramt+an+Gymnasien/Modulhandbuch+M_Ed_Lehramt+an+Gymnasien+2022-p-19444.pdf
38	Two-Subject-B. M.Ed.	https://www.physik.uni-kiel.de/de/studium/bama/Modulhandbuch%20Physik_2Faecher_BA_160721.pdf https://www.physik.uni-kiel.de/de/studium/bama/modulhandbuch-physik-med-01042021.pdf
39	Staatsexamen	https://friedolin.uni-jena.de/download/modulkataloge/de/23_128_phys_kf_2022.pdf

References

1. Funamori, M. The Impact of the Digital Age on Higher Education: Beyond Transformation from the Physical Space to the Digital Sphere. In Proceedings of the 5th IEEE IIAI International Congress on Advanced Applied Informatics (IIAI-AAI), Kanazawa, Japan, 10–14 July 2016. [CrossRef]
2. Mattila, A. The future educator skills in the digitization era Effects of technological development on Higher Education. In Proceedings of the 5th IEEE International Conference on e-Learning (econf), Manama, Bahrain, 18–20 October 2015. [CrossRef]
3. Telukdarie, A.; Munsamy, M. Digitization of Higher Education Institutions. In Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Macao, China, 15–18 December 2019. [CrossRef]
4. Dobrovska, D.; Andres, P. Digitization and Current Educational Changes in Switzerland—Inspiration for the Czech Republic? In Proceedings of the 22nd International Conference on Interactive Collaborative Learning (ICL)/48th IGIP International Conference on Engineering Pedagogy—Impact of the 4th Industrial Revolution on Engineering Education, Bangkok, Thailand, 25–27 September 2019; Springer International Publishing Ag: Cham, Switzerland, 2019. [CrossRef]
5. Michala, M.; Alexakos, C.; Tsolis, D. Digitization and Gamification Technologies in The Design, Development and Evaluation of a Digital Secondary Education Program for Greek Art and Culture. In Proceedings of the 10th International Conference on Education and New Learning Technologies (EDULEARN), Palma, Spain, 2–4 July 2018; Iated-Int Assoc Technology Education & Development: Palma, Spain, 2018.
6. Hohenstein, F.; Zimmermann, F.; Kleickmann, T.; Köller, O.; Möller, J. Have the education standards for teacher training programmes arrived in the university curriculum? *Z. Erzieh.* **2014**, *17*, 497–507. [CrossRef]
7. Kuske, S.; Holmberg, C.; Wensing, M.; Reuschenbach, B.; Büscher, A.; Neugebauer, E.; Pfaff, H.; Karbach, U.; Balzer, K.; Köpke, S.; et al. Educational Programmes in Health Services Research in Germany: Current State and Future Perspectives. *Gesundheitswesen* **2022**, *84*, 227–241. [CrossRef] [PubMed]
8. Ullrich, C.; Mahler, C.; Forstner, J.; Szecsenyi, J.; Wensing, M. Teaching implementation science in a new Master of Science Program in Germany: A survey of stakeholder expectations. *Implement. Sci.* **2017**, *12*, 8. [CrossRef] [PubMed]
9. Mikeska, J.N.; Shattuck, T.; Holtzman, S.; McCaffrey, D.F.; Duchesneau, N.; Qi, Y.; Stickler, L. Understanding science teaching effectiveness: Examining how science-specific and generic instructional practices relate to student achievement in secondary science classrooms. *Int. J. Sci. Educ.* **2017**, *39*, 2594–2623. [CrossRef]
10. Parker, J.; Heywood, D. Exploring the relationship between subject knowledge and pedagogic content knowledge in primary teachers' learning about forces. *Int. J. Sci. Educ.* **2000**, *22*, 89–111. [CrossRef]

11. Chrappan, M.; Bencze, R. Secondary School Students' Attitudes towards Science Subjects. In Proceedings of the 9th International Conference on Education and New Learning Technologies (EDULEARN), Barcelona, Spain, 3–5 July 2017; Iated-Int Assoc Technology Education & Development: Barcelona, Spain, 2017.
12. Peter, C.; Sprenger, S. Digitalization and Geography Education A Curriculum Analysis. *Erdkunde* **2022**, *76*, 3–19. [CrossRef]
13. Serbanescu, L. Ability To Use Technical Audio-Visual Means and Information And Communication Technology (ICT)—A Fundamental Requirement in Teachers' Didactic Training. In Proceedings of the 9th International Conference eLearning and Software for Education, Bucharest, Romania, 25–26 April 2013; Carol I Natl Defence Univ Publishing House: Bucharest, Romania, 2013.
14. Dagiene, V. Teaching information technology and elements of informatics in lower secondary schools: Curricula, didactic provision and implementation. In Proceedings of the 3rd International Conference on Informatics in Secondary Schools, Evolution and Perspectives (ISSEP 2008), Torun, Poland, 1–4 July 2008; Springer: Berlin, Germany, 2008.
15. Midoro, V. How teachers and teacher training are changing. In *International Conference on the Bookmark of the School of the Future*; Kluwer Academic Publishers: Vina Del Mar, Chile, 2000.
16. Hudson, B. Didaktik Design for Technology Supported Learning. *Z. Erzieh.* **2008**, *10*, 139–157.
17. Starčič, A.I.; Cotic, M.; Solomonides, I.; Volk, M. Engaging preservice primary and preprimary school teachers in digital storytelling for the teaching and learning of mathematics. *Br. J. Educ. Technol.* **2016**, *47*, 29–50. [CrossRef]
18. Körtesi, P.; Simonka, Z.; Szabo, Z.K.; Guncaga, J.; Neag, R. Challenging Examples of the Wise Use of Computer Tools for the Sustainability of Knowledge and Developing Active and Innovative Methods in STEAM and Mathematics Education. *Sustainability* **2022**, *14*, 23. [CrossRef]
19. Krasna, M.; Korze, D.; Kaucic, B. Searching for the reasons why ICT is not adequately used in schools. In Proceedings of the 41st IEEE International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 21–25 May 2018.
20. Loudova, I. Continuing Professional Development in ICT for Primary School Teachers, Reflections and Issues. In Proceedings of the 13th International Conference on Blended Learning (ICBL), Bangkok, Thailand, 24–27 August 2020; Kasetsart Univ, ELECTR NETWORK; Springer International Publishing Ag: Bangkok, Thailand, 2020. [CrossRef]
21. Luis, A.R. ICT Training and Pre-service Teachers: Embracing the Challenge at Master's Level. In Proceedings of the World Conference on Information Systems and Technologies (WorldCIST), Madeira, Portugal, 11–13 April 2017; Springer: Berlin, Germany, 2017. [CrossRef]
22. European Commission; Joint Research Centre; Redecker, C.; Punie, Y. *European Framework for the Digital Competence of Educators: DigCompEdu*; Punie, Y., Ed.; Publications Office: Copenhagen, Denmark, 2017; Available online: https://data.europa.eu/doi/10.2760/159770 (accessed on 23 March 2023).
23. Thyssen, C.; Huwer, J.; Irion, T.; Schaal, S. From TPACK to DPACK: The Digitality-Related Pedagogical and Content Knowledge”-Model in STEM-Education. *Educ. Sci.* **2023**, *13*, 769.
24. Huwer, J.; Irion, T.; Kuntze, S.; Schaal, S.; Thyssen, C. Digitality-related Paedagogical Content Knowledge (DPaCK)—A Framework for teacher education in the digital age. In *Education Research Highlights in Mathematics, Science and Technology*; IRES Publishing: Iowa, IA, USA, 2019; pp. 298–309.
25. Mishra, P.; Koehler, M.J. Technological pedagogical content knowledge: A framework for teacher knowledge. *Teach. Coll. Rec.* **2006**, *108*, 1017–1054. [CrossRef]
26. Koehler, M.J.; Mishra, P. Introducing TPCK. In *The Handbook of Technological Pedagogical Content Knowledge (TPCK) for Educators*; AACTE Committee on Innovation and Technology, Ed.; Lawrence Erlbaum Associates: Mahwah, NJ, USA, 2008; pp. 3–29.
27. Benz, G.; Ludwig, T. Going beyond general competencies in teachers' technological knowledge: Describing and assessing pre-service physics teachers' competencies regarding the use of digital data acquisition systems and their relation to general technological knowledge. *Front. Educ.* **2023**, *8*, 1180973. [CrossRef]
28. Freese, M.; Teichrew, A.; Winkelmann, J.; Erb, R.; Ullrich, M.; Tremmel, M. Measuring teachers' competencies for a purposeful use of augmented reality experiments in physics lessons. *Front. Educ.* **2023**, *8*, 1180266. [CrossRef]
29. Thoms, L.-J.; Colberg, C.; Heiniger, P.; Huwer, J. Digital Competencies for Science Teaching: Adapting the DiKoLAN Framework to Teacher Education in Switzerland. *Front. Educ.* **2022**, *7*, 802170. [CrossRef]
30. Becker, S.; Bruckermann, T.; Finger, A.; Huwer, J.; Kremser, E.; Meier, M.; Thoms, L.-J.; Thyssen, C.; von Kotzebue, L. *Orientierungsrahmen Digitale Kompetenzen für das Lehramt in den Naturwissenschaften–DiKoLAN. Digitale Basiskompetenzen–Orientierungshilfe und Praxisbeispiele für die universitäre Lehr-amtsausbildung in den Naturwissenschaften*; Joachim Herz Stiftung: Hamburg, Germany, 2020; pp. 14–43.
31. Kotzebue, L.v.; Meier, M.; Finger, A.; Kremser, E.; Huwer, J.; Thoms, L.-J.; Becker, S.; Bruckermann, T.; Thyssen, C. The Framework DiKoLAN (Digital Competencies for Teaching in Science Education) as Basis for the Self-Assessment Tool DiKoLAN-Grid. *Educ. Sci.* **2021**, *11*, 775. [CrossRef]
32. Aumann, A.; Schnebel, S.; Weitzel, H. The EnTPACK Rubric: Development, Validation, and Reliability of an Instrument for Measuring Pre-Service Science Teachers' Enacted TPACK. *Front. Educ.* **2023**, *8*, 1190152. [CrossRef]
33. Henne, A.; Möhrke, P.; Thoms, L.-J.; Huwer, J. Implementing Digital Competencies in University Science Education Seminars Following the DiKoLAN Framework. *Educ. Sci.* **2022**, *12*, 356. [CrossRef]

34. Baumann, B.; Groß, J.; Michelsen, M. An app by students for students—The DPaCK-model for a digital collaborative teamwork project to identify butterflies. *Front. Educ.* **2023**, *8*, 1190462. [CrossRef]
35. Braun, D.; Huwer, J. Computational literacy in science education—A systematic review. *Front. Educ.* **2022**, *7*, 937048. [CrossRef]
36. Fagerlund, J.; Häkkinen, P.; Vesisenaho, M.; Viiri, J. Computational thinking in programming with Scratch in primary schools: A systematic review. *Comput. Appl. Eng. Educ.* **2021**, *29*, 12–28. [CrossRef]
37. Trujillo Torres, J.; Gómez García, G.; Ramos Navas-Parejo, M.; Soler Costa, R. The development of information literacy in early childhood education teachers: A study from the perspective of the education center's character. *J. Technol. Sci. Educ.* **2020**, *10*, 47–59. [CrossRef]
38. Wing, J. Computational thinking. *Commun. ACM* **2006**, *49*, 33–35. [CrossRef]
39. Voogt, J.; Fisser, P.; Good, J.; Mishra, P.; Yadav, A. Computational thinking in compulsory education: Towards an agenda for research and practice. *Educ. Inf. Technol.* **2015**, *20*, 715–728. [CrossRef]
40. Grover, S.; Pea, R. Computational Thinking in K–12. *Educ. Res.* **2013**, *42*, 38–43. [CrossRef]
41. Rubinstein, A.; Chor, B. Computational thinking in life science education. *PLoS Comput. Biol.* **2014**, *10*, e1003897. [CrossRef] [PubMed]
42. Swanson, H.; Anton, G.; Bain, C.; Horn, M.; Wilensky, U. Introducing and Assessing Computational Thinking in the Secondary Science Classroom. In *Computational Thinking Education*; Kong, S.-C., Abelson, H., Eds.; Springer: Berlin/Heidelberg, Germany, 2019; pp. 99–117.
43. Tang, X.; Yin, Y.; Lin, Q.; Hadad, R.; Zhai, X. Assessing computational thinking: A systematic review of empirical studies. *Comput. Educ.* **2020**, *148*, 103798. [CrossRef]
44. Diethelm, I.; Goschler, J. Questions on spoken language and terminology for teaching computer science. In Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education, Vilnius, Lithuania, 4–8 July 2015; ACM Digital Library: New York, NY, USA, 2015; pp. 21–26.
45. Dengel, A.; Heuer, U. A curriculum of computational thinking as a central idea of information & media literacy. In Proceedings of the 13th Workshop in Primary and Secondary Computing Education, Potsdam, Germany, 4–6 October 2018; pp. 1–6.
46. Chongtay, R.; Robering, K. Computational literacy, a layered approach for the humanities. In *ICERI2016 Proceedings*; IATED: New York, NY, USA, 2016; pp. 551–561.
47. Djambong, T. Computational Thinking in Connection with the Acquisition of Key Skills for the 21st Century: Reflection from Literature Review. In *EDULEARN16 Proceedings*; IATED: New York, NY, USA, 2016; pp. 500–509.
48. Guzdial, M. Computing education as a foundation for 21st century literacy. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education, Minneapolis, MN, USA, 22 February–2 March 2019; pp. 502–503.
49. Reffay, C.; Viroonluecha, P. Computational Thinking Nurturing Skills and Inspiring Pedagogy for Sustainable Education in the 21st Century. In Proceedings of the Sustainable ICT, Education and Learning: IFIP WG 3.4 International Conference, SUZA 2019, Zanzibar, Tanzania, 25–27 April 2019; Revised Selected Papers 1. Springer International Publishing: Cham, Switzerland, 2019; pp. 66–77.
50. Martins-Pacheco, L.H.; Degering, L.; Mioto, F.; von Wangenheim, C.; Borgato, A.; Petri, G. Improvements in bASES21: 21st-century skills assessment model to K12. In Proceedings of the 12th International Conference on Computer Supported Education, Prague, Czech Republic, 2–4 May 2020; pp. 297–307.
51. Röhner, G.; Brinda, T.; Fricke, M.; Gevers, M.; Hug, A.; Losch, D.; Puhlmann, H. *Gemeinsamer Referenzrahmen Informatik (GeRRI)–Mindeststandards für Die auf Informatik Bezogene Bildung*; Gesellschaft für Informatik e.V.: Bonn, Germany, 2020.
52. Puentedura, R. Substitution, augmentation, modification, and redefinition (SAMR) model. *Indones. J. Inform. Educ.* **2006**, *7*, 8–17.
53. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ* **2021**, *372*, n71. [CrossRef]
54. Roepke, R.; Nell, M.; Schroeder, U. *Semi-Assisted Module Handbook Content Extraction for the Application of Curriculum Analytics*; Gesellschaft für Informatik e.V.: Bonn, Germany, 2023.
55. Föll, P.; Thiesse, F. Exploring Information Systems Curricula: A Text Mining Approach. *Bus. Inf. Syst. Eng.* **2021**, *63*, 711–732. [CrossRef]
56. Koch, J.; Koch, J.; Sträßner, M.; Coners, A. Theory and Practice-What, with What and How Is Business Process Management Taught at German Universities? In Proceedings of the International Conference on Business Process Management, Münster, Germany, 11–16 September 2022; Springer International Publishing: Cham, Switzerland, 2022; pp. 34–39.
57. Dietrich, H.; Zhang, Y.; Klopp, E.; Brünken, R.; Krause, U.-M.; Spinath, F.M.; Stark, R.; Spinath, B. Scientific competencies in the social sciences. *Psychol. Learn. Teach.* **2015**, *14*, 115–130. [CrossRef]
58. Ullah, S.; Kimani, D.; Bai, Y.; Ahmed, R. Assessing the design of accounting modules across UK higher educational institutions. *Cogent Bus. Manag.* **2018**, *5*, 1510717. [CrossRef]
59. Iraci, D.; Urzelai, B.; Dance, G. Team Coaching Practice in the UK, a Comparative Study. In *Team Academy in Diverse Settings*; Routledge: London, UK, 2022; pp. 5–18.
60. Choate, B.; Dubosarsky, M.; Chen, K. An innovative professional development model for teaching robotics to novice educators. In Proceedings of the 2018 IEEE Frontiers in Education Conference (FIE), San Jose, CA, USA, 3–6 October 2018; pp. 1–5.

61. Gunčaga, J.; Kopczyński, T. Supporting mathematical and digital competences useful for STEM education. In *E-Learning and STEM Education*; Uniwersytet Śląski: Katowice, Poland, 2019.
62. Juškevičienė, A.; Dagienė, V.; Dolgopolovas, V. Integrated activities in STEM environment: Methodology and implementation practice. *Comput. Appl. Eng. Educ.* **2021**, *29*, 209–228. [CrossRef]