Chapter 4
Students' Activities from Elementary to High School

Rukiye Altin, Ozge Mísírlí, Irene Zanardi, Claudia Maria Cutrupi, Sunny K. O. Miranda, Valentina Dagiene, and Monica Landoni

4.1 Personas

This chapter represents recruitment initiatives, which might be useful for different kind of personas. Thus before those initiatives are unfolded, two representatives that might find them useful, a school principal, named imaginatively Alex, and a school teacher, named Jamila, are described below.

R. Altin (✉)
Kiel University, Kiel, Germany
e-mail: ral@informatik.uni-kiel.de

O. Mísírlí
Eskisehir Osmangazi University, Eskisehir, Turkey

I. Zanardi · M. Landoni
Università della Svizzera italiana, Lugano, Switzerland
e-mail: irene.zanardi@usi.ch; monica.landoni@usi.ch

C. M. Cutrupi
Norwegian University of Science and Technology, Trondheim, Norway
e-mail: claudia.m.cutrupi@ntnu.no

S. K. O. Miranda
University of Coimbra, Coimbra, Portugal
e-mail: sunnymiranda@dei.uc.pt

V. Dagiene
Vilnius University, Institute of Data Science and Digital Technologies, Vilnius, Lithuania
e-mail: valentina.dagiene@mif.vu.lt

© The Author(s) 2025

B. Penzenstadler et al. (eds.), *Actions for Gender Balance in Informatics Across Europe*, https://doi.org/10.1007/978-3-031-78432-3_4

Alex, the school principal. Looking at the resources in this chapter, Alex the school principal takes the recommendations to their next school board meeting. Some of the teachers are enthusiastic about developing role models in the classroom, others are a bit more sceptical. So Alex starts a group activity where the teachers pair up with a partner, interview each other on what role models they have in their lives, and create role model descriptions for each other.

Jamila, school teacher. Looking at the resources in this chapter, Jamila is a school teacher has passion about STEM and wants to support their students being open to that direction. She needs suggestions about how to raise awareness on gender diversity in computer science so Jamila starts a group activity where the teachers can rainstorm about inspiring activities for students from kindergarten to high school.

4.2 Introduction

In this chapter, We focus on school-based activities. We explain the experience of teaching Informatics in elementary to high schools, whether viewed as a specific or transversal subject and discuss the impact of gender differences on students' and instructors' attitudes toward it.

Recently, primary, middle and high schools have been striving to integrate Informatics into their curricula, recognizing the increasing demand for programming and algorithmic thinking skills in the future job market. Educating students in Informatics from a young age is essential because it naturally enhances various skills, including problem-solving, creative thinking, and algorithmic/computational thinking, thereby preparing students for diverse fields of study and careers. Researchers assert that, given Informatics' potential to become central to a technologically advanced society, every student should have the opportunity to engage with it [73].

Promoting computing education for primary and middle school students involves implementing engaging, age-appropriate activities that stimulate interest and understanding of fundamental computing concepts. Effective education in Informatics requires considering students' learning levels, employing effective pedagogical approaches, and incorporating interdisciplinary, real-life collaborative examples [51, 53].

Despite the increasing incorporation of Informatics in primary and middle schools and its growing popularity in academia, girls remain underrepresented. To achieve gender diversity in Informatics, it is essential to encourage girls from an early age to pursue this field [73]. Certain activities are particularly effective for teaching computer science to girls. These activities are designed not only to convey computer science concepts but also to engage and inspire girls, fostering an inclusive and stimulating learning environment. Effective approaches include hands-on programming and robotics experiences, narrative-driven and project-based learning, game playing and design, interdisciplinary projects, long-term school-based interventions, and strong support systems [73]. We start by reporting on a long-term study that aims to find out what children think a software engineer looks like to better understand children's mental models and the role played by gender. Then, the following section provides suggestions for effective types of activities widely recognised in academic literature organised according to whether these are to be administered in elementary or high school. We present various after-school programs divided according to suitable age, that are available to children to help them understand the possibilities of Informatics in a playful environment. By implementing these interventions in schools, we believe that it is possible to foster a sustained interest in IT among girls and contribute to reducing the gender gap in the IT field, whether it is viewed as a specific or a transversal subject, and we discuss the impact of gender differences in students' and instructors' attitudes toward it. In the last section, we present the various after-school programs

! Attention

In this chapter, we will talk about elementary, middle, and high schools. We are aware that there are national variations in these designations. Here, we are talking about kids in elementary school, who are typically six to eleven years old; middle schoolers, who are typically twelve to fourteen years old; and high schoolers, who are typically fifteen to eighteen years old.

4.3 Draw a Software Engineer

Perhaps because it is not usual to come across a woman computer scientist in college or the workplace, one may wonder what the secret is to the successful career paths of these women. How did they arrive here? And why did they stay when others left? Some have followed a straight path, albeit not without challenges, drawn by personal interest. Others moved into computing after studying and working in other fields, owing to the multidisciplinary nature that technology's ubiquity brings. Regardless of the route taken, self-reliance and perseverance have been critical. In fact, one reason why women do not stay in computing is the difficulties of working in a field perceived as masculine [8, 31, 35, 49]. These challenges have little to do with skills, talents, or predispositions, but rather with the ingrained gendered perception of the field [49]. This perception is distorted by stereotypes and mental models, which are important decision-making items, particularly regarding career choices. Our childhood aspirations heavily influence our interests, and in turn, they are influenced by what we deem to be socially appropriate and rewarding [34]. As a result, when an activity is perceived as masculine, those who do not align with masculine characteristics can almost hear a quiet warning that it is not for them.

Before sharing information about school and after-school activities, we examine children's perceptions of people working in the computing sector to determine which stereotypes exist and when they emerge.

4.3.1 Aspirations and Gender in Children

Among the models that explain how we form aspirations, Gottfredson proposed the theory of circumscription and compromise [34], where they identified socioeconomic status, ability, and gender as constructs of career aspiration. **The perception of job-self compatibility is specifically linked to self-concept and occupational images** (Fig. 4.1). The self-concept is one's perception of who they are, who they want to be, and who they are not. As a result, it is influenced by factors such as social class, intelligence, interests, values, personality, and even one's social standing. More interestingly to the current topic, the self-concept includes one's gender, as in its individual cultural dimension gender influences the beliefs one holds about oneself [61]. Occupational images are generalizations about a profession, such as what the job entails, how prestigious it is, and what personal characteristics are appropriate for the worker. Also in this case, gender plays a role, but this time we have to consider its interactional cultural dimension [61]: gender stereotypes, cognitive biases, and expectations shape our understanding of the world and influence how we conceptualize things.

Let us look at two examples. If we see ourselves as feminine and caring, and we see nurses as feminine-typed and caring, there will be an alignment, resulting in a high perception of job-self compatibility, which will lead to a job preference. If there is a high level of compatibility, low barriers, and abundant opportunities, the occupation is socially acceptable. If we regard the nurse title as prestigious, nursing can become our occupational aspiration. In this case, nursing would be classified as

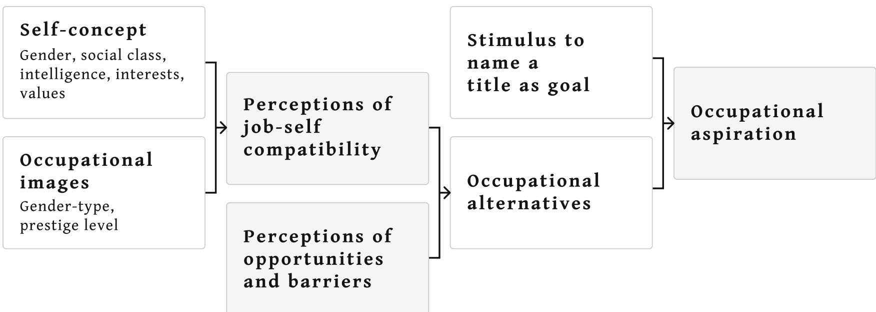

Fig. 4.1 Constructs relations in theory of circumscription and compromise, adapted from [33]

highly feminine and quite prestigious on our cognitive map of occupations, which develops along prestige and gender typing. If we consider computing, the outcome would be different.

Important

For someone to be interested in computing as a possible career field, there must be a correspondence with the characteristics they believe are required for the field: computing is perceived as masculine and thus has masculine characteristics, which may act as a barrier to people with more feminine attributes.

To delve deeper into Gottfredson's theory, the process of circumscription and compromise of aspirations begins in childhood and progresses through four stages. The first three stages focus on excluding unacceptable options (circumscription), while the fourth stage focuses on selecting the most feasible occupations from among the preferred ones (compromise) (Fig. 4.2). The first stage begins in preschool, when children recognize occupations as roles and stop expressing their desire to become animals or fantasy characters. In the second stage, from six to eight years old, children categorize people as good or bad, with this distinction primarily guided by what is appropriate (good) for one's gender. They begin categorizing activities into feminine and masculine categories. This aligns with the timeline proposed by Bem's theory of gender schemata: children hold mental representations about what is feminine and what is not, and perform their gender based on their position on the spectrum [6]. Then, in the third stage, from nine to thirteen, they assess the prestige of the occupation, recognizing that there is a hierarchy in terms of professional status. They have now a variety of potential careers that they will rank in the fourth stage.

Gottfredson's theory of circumscription and compromise makes it easy to see that **gender matters and plays a significant role early on in the process of selecting an aspiration**. Not surprisingly, kids who identify more strongly with one gender tend to have more gender-typical goals [25]. Aspirations can be classified based on gender, depending on whether the job subjects are things or people. By categorizing occupations into people-oriented and things-oriented, we can see that women prefer the first category and men prefer the second [26, 74]. Informatics is considered a

Fig. 4.2 Stages of circumscription and compromise on aspirations

things-oriented occupation [37], with no human contact [10], making it look more compatible with boys.

Interestingly, Gottfredson points out that **the narrowing of options must be considered irreversible, because rejected options are not reconsidered spontaneously; rather, children reconsider them only when prompted to do so**. This emphasizes the importance of first understanding the computer scientist's mental models, or occupation images, as defined by Gottfredson, and then taking action to avoid rejecting or reconsidering this career. While there are few actions we can take to improve a person's self-concept, we can change how the field is presented to demonstrate how much variety exists and, as a result, open up to more traditionally feminine characteristics. One way to investigate the occupational image that children hold of a profession is with drawings.

4.3.2 Years of Drawing Occupational Images

Drawings can aid in visualizing children's implicit associations when thinking about a certain thing. In some ways, asking to draw a professional figure gives us insight into that figure's identity: what they look like, what they wear, what they do, and how they do it. Moreover, because we are talking about children, drawing has the advantage of being more accessible and less cognitively demanding than verbal description, which would still require some awareness of the associations.

Drawing as a tool was initially used to investigate the scientist, when Chalmers et al. [17] developed the Draw a Scientist Test (DAST), an art-based tool for gathering images of STEM professionals. Because of its ease of use and flexibility, this tool has been widely used over the years to highlight the presence of stereotypical elements in drawing representations. Given that the task is the same, it is possible to track potential shifts in participants' perceptions as our society changes. Furthermore, the DAST task's ease of translation promotes cross-national research. As a result, it is feasible to compare drawings from various countries and investigate contrasts and similarities that may be influenced by cultural differences [30, 52]. In Informatics, these investigations focused on the computer scientist (DACST), the programmer (DAPT), and the computer user (DACUT) (Fig. 4.3).

We can trace the history of children's perceptions of people in Informatics since 1994 [9] (Fig. 4.4). MacOS Classic and Windows have been around for about ten years, with their innovative graphical user interfaces; Tim Berners-Lee published the first website three years prior, and Mosaic, the first web browser, was released to the public one year earlier. The year 1994 saw an impressive number of developments. The new web browser Netscape popularized the web, which became free of commercial restrictions and increasingly interoperable thanks to the World Wide Web Consortium (W3C), which was founded the same year. Yahoo! and Amazon were founded, and PlayStation appeared to revolutionize videogames, which later became an industry closely linked to boys' programming ambitions. Approximately 20 million people were online [40]. In this exciting time for computing, Barba

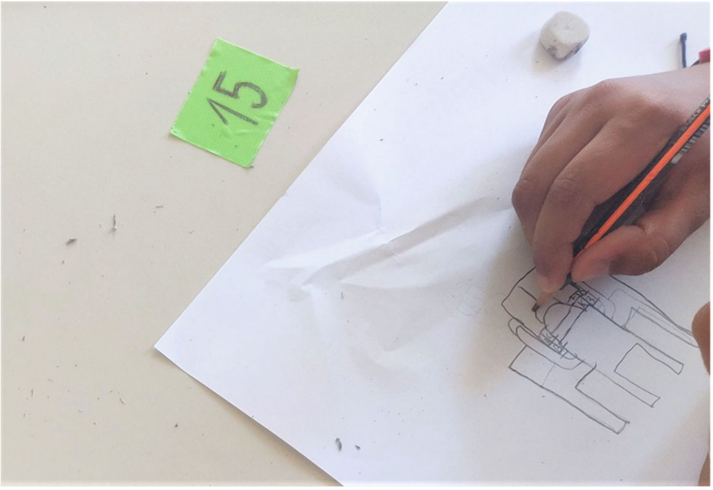

Fig. 4.3 Child drawing a software engineer

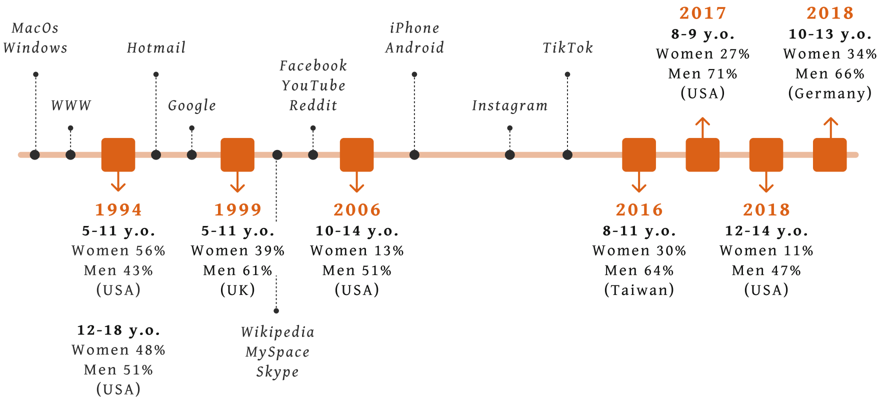

Fig. 4.4 Timeline of depiction of women and men as people working with computers

et al. asked children to draw someone they thought was a computer user. While elementary school children provided no stereotypical representation, from grade 7 onwards, an increasing number of drawings depicted them as a "nerd". According to the teenagers, nerds are thin and pale teenage boys who part their hair in the middle, wear eyeglasses, carry mechanical pencils in their pockets, and dress in oversized clothes and high-waters trousers. Interestingly, elementary school children drew more women as computer users, whereas middle and high school children drew more men.

Five years later, in 1999, Google and Hotmail were already established, Windows had progressed from W95 to W98, Apple had introduced the iMac, Wi-Fi had revolutionized internet connectivity, and Napster had only recently appeared. In just five years, the users grew from 20 million to 280 million [40]. Despite the growing popularity of computers, men did not account for the vast majority of drawings (60%). However, boys were less likely to draw women than girls, with 30% of girls drawing a man and only 4% of boys drawing a woman [11]. This means that computers were perceived as “boys’ things” by boys.

By 2006, the biggest web players entered the arena, and the number of online users was over 1 billion [40]. In chronological order, Wikipedia, LinkedIn, MySpace, WordPress, Skype, Facebook, Gmail, YouTube, GoogleMaps, Reddit, and Twitter all made their debuts laying the groundwork for the modern online world. In this context, computer users’ occupational images included their gender (men) and eyeglasses. Some individuals exhibited negative social characteristics and abnormal body weight, which are associated with the nerd stereotype [48]. Similarly to the previous study, men are depicted in slightly more than half of all drawings. However, also in this case, boys were less likely to draw a woman (6th grade 10%, 8th grade 3%) than girls a man (6th grade 53%, 8th grade 67%).

Ten years later, online users tripled [40] and too many advancements should be listed. A few notable events include the release of iPhone in 2007 and Android in 2008, the founding of Instagram in 2010, and the launch of TikTok in 2016. In 2016, 1.5 billion smartphones were sold, up from 122 million in 2007 [67]. Between 2016 and 2018, four studies looked into the occupational images of people in computing. In 2016 in Taiwan [16], elementary school children drew people in computer science smiling, wearing glasses, a hat and a tie, and carrying a toolbox. Approximately 73% of computer scientists were men. Similarly, in the USA in 2017 [35], men were depicted more often than women (71%). Fewer stereotypes were present, the main ones were about eyeglasses and baldness. The following year, still in the USA [58], middle school students drew more men than women (80%), depicting them with eyeglasses. At the same time, in Germany, children’s perception was the same [12]: more men were depicted and boys were more definitely less likely to draw a woman than girls drawing a man.

There were no significant changes in the depicted gender, though a slight improvement can be seen. Four years after the previous results, and following the 2019 pandemic, we proposed a new study to determine whether anything happened.

4.3.3 How Informatics Is Imagined Now

Through the EUGAIN network, an empirical study was conducted using the Draw-A-Picture method to investigate children’s perceptions of software engineering figures [21], called Draw a Software Engineer test (DASET). Specifically, the study aimed to understand how children perceive software engineers and the stereotypical elements in their representations, examining variations based on gender and age.

The investigation was conducted in Milan, Italy, involving 371 elementary school students from diverse socioeconomic and cultural backgrounds. The sample included 192 girls, 166 boys, and 13 children identifying as “other,” with an average

age of around 8 years. Participation was voluntary, with parental consent obtained beforehand. The children's gender identification and grade level were recorded, while other information was kept anonymous.

During the activity, children were asked to draw a person working in the software engineering field (Fig. 4.3), ensuring the terminology used did not influence the result. Before starting the activity, the children were asked if they knew what software engineering was, with children associating it with concepts like computers, hackers, videogames, and robots. To maintain engagement, the activity included a storytelling element featuring a human-controlled robot named CYB, which responded to the drawings, providing a rewarding interaction for the children.

The sessions, conducted in groups of 15–20 children, lasted 30 minutes, with clear instructions and measures to avoid peer influence on the results. Children used pencils and markers, and their drawings were analyzed based on a coding scheme developed from existing literature [17]. This scheme included several parameters such as the validity of the drawings, generalities like gender and skin color, appearance (beard, clothes, eyeglasses, etc...), environment, and technical equipment. The coding was performed by three authors and the findings of the study were published in a full paper at ICSE 2023, in Software Engineering in Society track [21].

At first glance, compared to the occupational images depicted in the past, there are no big and exciting differences (Table 4.1). Men appeared slightly more often than women in children's drawings, and the only traditional stereotype was eyeglasses, but in less than a fourth of the cases. We can also add that non-binary children included in the study were aligned with girls and boys. What is interesting, though, is that the depicted gender was balanced for girls, boys, and non-binary children alike. We saw earlier that while around half of the girls used to portray men, boys were extremely skewed and did not draw women. That showed a lack of open-mindedness and a clear typing of programming and computers as exclusively masculine. With the years, it seems that something changed: almost half of the boys drew a woman. Informatics then is less masculine and more egalitarian (Figs. 4.5 and 4.6). The hope is that it is becoming so common and fundamental in our society that it is not a man's prerogative anymore, but everyone should be concerned with it.

Fig. 4.5 Drawing of women in software engineering

Fig. 4.6 Drawing of men in software engineering

The occupational images we collected were analyzed in terms of tools and environment to get a glimpse of the context children imagine for the software engineer (Table 4.1). Usually, the person is at the desk either standing up or sitting on a chair. Rarely there are references to videogames and science and almost no signs of junk food and coffee. Usually, there is a computer or a robot, sometimes a keyboard. Appearance-wise, the software engineer has normal hair and rarely a beard; no acne or other elements against beauty standards, and no geek shirts. In the end, the person who works in computing is not seen as a "nerd" as described in the first study of 1994.

If we look at age differences, we can also observe another interesting detail, even though less encouraging. The older the children were, the more men were depicted (Table 4.1). Not only that, but drawings contained more details: contextual elements were richer. This seems to point out that they got to know better what Informatics was and with that knowledge, they began to associate it with masculinity. This shift, which is still not dramatic, can be especially seen between grades 4 and 5.

4.3.4 How to Apply DASET

As it was mentioned above, one of the interesting characteristics of this type of study is its nature of replication. However, in order to ensure that results from different investigation are comparable, it is necessary to draw replication patterns. In the following, we will present the lessons learned from our research and provide information for those who wish to conduct the same experiment.

The present content was also presented during a workshop with EUGAIN network members and conducted by the authors of the paper [21].

Tips

To avoid reinforcing stereotypes, you can frame the activity as a collaborative effort to assist a robot in learning how to draw. This approach fosters a neutral and inclusive environment, encouraging children to freely express their ideas without regard for gender stereotypes.

Table 4.1 Parameters percentages

Parameter	Other (NB) (n = 13)	Girls (n = 192)	Boys (n = 166)	Total (n = 371)
Man Software Engineer	P = 0.54	P = 0.46	P = 0.56	P = 0.51
Woman Software Engineer	P = 0.46	P = 0.48	P = 0.39	P = 0.44
No gender Software Engineer	P = 0.00	P = 0.06	P = 0.05	P = 0.05
Pink skin	P = 0.54	P = 0.66	P = 0.60	P = 0.63
Brown skin	P = 0.00	P = 0.00	P = 0.00	P = 0.00
Unshaded skin	P = 0.46	P = 0.34	P = 0.40	P = 0.37
White lab coat	P = 0.00	P = 0.04	P = 0.05	P = 0.04
Elegant clothes	P = 0.00	P = 0.02	P = 0.04	P = 0.03
Geek shirt	P = 0.00	P = 0.04	P = 0.02	P = 0.03
Glasses	P = 0.15	P = 0.15	P = 0.19	P = 0.17
Beard	P = 0.08	P = 0.05	P = 0.08	P = 0.06
Acne	P = 0.00	P = 0.00	P = 0.01	P = 0.01
Crazy hair	P = 0.08	P = 0.03	P = 0.03	P = 0.03
Desk	P = 0.69	P = 0.58	P = 0.57	P = 0.58
Chair	P = 0.23	P = 0.29	P = 0.27	P = 0.27
Office chair	P = 0.00	P = 0.04	P = 0.11	P = 0.07
Bookshelf	P = 0.15	P = 0.02	P = 0.02	P = 0.02
Production machine	P = 0.00	P = 0.04	P = 0.06	P = 0.02
Secrecy symbols	P = 0.00	P = 0.03	P = 0.01	P = 0.01
Toys	P = 0.00	P = 0.01	P = 0.02	P = 0.01
Videogames	P = 0.08	P = 0.02	P = 0.06	P = 0.04
Food	P = 0.00	P = 0.01	P = 0.01	P = 0.01
Drinks	P = 0.08	P = 0.03	P = 0.03	P = 0.03
Computer	P = 0.62	P = 0.54	P = 0.52	P = 0.54
Robot	P = 0.31	P = 0.45	P = 0.37	P = 0.41
Earphones	P = 0.08	P = 0.01	P = 0.06	P = 0.03
Keyboard	P = 0.08	P = 0.16	P = 0.16	P = 0.16
Mouse	P = 0.00	P = 0.07	P = 0.07	P = 0.07
Smartphone	P = 0.00	P = 0.07	P = 0.11	P = 0.08
Tablet	P = 0.00	P = 0.02	P = 0.02	P = 0.02
Console	P = 0.00	P = 0.01	P = 0.04	P = 0.02
Controller	P = 0.00	P = 0.05	P = 0.04	P = 0.04
TV	P = 0.08	P = 0.03	P = 0.01	P = 0.02
Laboratory tools	P = 0.00	P = 0.02	P = 0.01	P = 0.01
Mechanic tools	P = 0.08	P = 0.05	P = 0.02	P = 0.04

Specify the Task To start the activity, the researchers need to present the tasks that are essentially required for the activity. At this point, the researchers can ask the group if they know who a software engineer is. We urge not to suggest any kind of association. It is possible that their understanding includes details not pertaining to the software engineer alone. In this case, it is important to take note of it.

Choose a Gender-Neutral Language When delivering the main instruction, it is critical to avoid using gendered language, if present, ensuring that the guidance remains neutral and inclusive.

Plan the Questionnaire Finally, we recommend that the children write a unique code on their drawing paper. This code will also be included in the survey to ensure a connection between their drawing and responses while maintaining anonymity. The team can collect the drawings after 30 minutes. Before accepting the drawing, we recommend that the child report the code. To avoid any potential influence on the children's responses, the survey should be administered after the drawing activity. It is critical that the children be asked to enter the same code that they used for their drawings, as this will help maintain the link between their artwork and survey responses.

4.4 School-Based Activities

Informatics integrated into education should not only cover the fundamentals of computers and programming, but also promote a direct connection with twenty-first-century skills such as problem-solving, algorithmic thinking, creativity, and so on [19, 68].

The literature recommends the following activities as best practices for integrating Informatics into the school curriculum. The specific activities within each topic may vary depending on the curriculum, how teachers implement it, and the student's interests and needs.

Unplugged Activities
Topic: Computational Thinking Concepts, Algorithms
Description: Teaching about Algorithms, Events, Loops [68].
Activities:

1. Students might engage in activities involving sequencing steps to complete a task, such as writing instructions for a partner to follow or creating a flowchart for a simple process (e.g., making a sandwich).
2. Activities could involve identifying cause-and-effect relationships, simulating event-driven actions (e.g., clapping hands to trigger a specific response), or creating simple "if-then" scenarios.
3. Students might participate in activities that involve repeating instructions, such as creating a dance routine with repeating movements or simulating a loop using physical objects.

Visual Programming with Block-Based Languages
Topic: Programming, Problem-solving, Creativity
Description: Programming languages like Scratch and Blockly are highly recommended for girls as they offer a visual and intuitive introduction to programming. These tools remove the complexity of syntax, allowing students to focus on logic and design.
Activities:

1. Storytelling and Animation—Girls often show interest in narratives and stories. Tools that allow the creation of interactive stories or animations, such as Alice or Scratch, are effective for teaching programming concepts for girls [41, 51].
2. Creating interactive stories, animations, or games using visual programming blocks. This could involve designing characters and backgrounds and programming character movements, interactions, and dialogue.

Educational Robotics
Topic: Computational thinking, Programming, Robotics, Problem-solving, Creativity.
Description: Robotics can effectively introduce CS concepts to elementary school students and foster engagement with STEM [29]. The robotics programming course significantly improved computational thinking and creativity in students [56]. Girls demonstrated a significantly more significant improvement in creativity [56] and performed slightly better than boys in combining CS concepts during robot programming [51]. Students learn programming concepts by designing, building, and programming robots to perform specific tasks. Kits like LEGO Mindstorms, Bee-Bots, Arduino, Micro:bit, Makey Makey are popular choices.
Activities:

1. Use Scratch programming to control electronic components such as Arduino, Micro:bit, Makey Makey to create interactive projects that respond to physical inputs [36]. This could involve using everyday objects (e.g., fruits, toys) as conductive materials to control Scratch programs.
2. Use block-based language to teach basic programming concepts while students explore robot functionalities [56].
3. Use a multilanguage robot programming platform for students to discover new CS concepts [51].

Presenting Role Models
Topic: Women Encouragement and Inspiration, Fight Bias and Stereotypes.
Description: Exposing girls to women role models in computer science can significantly increase their interest and confidence in this field, besides awakening new perceptions of STEM careers and professionals [4].
Activities:

1. Promote activities to expose students to women STEM professionals and their work [13] This can be accomplished by showing videos, films or the presence of women STEM professionals at the school.
2. Create a booklet that features women role models and their innovative work in Computer Science. The creation of booklets gives visibility to women role models in Computer Science and makes girls feel inspired and motivated [5]CS4FN, 2024.

Serious Games
Topic: Fight Bias and Stereotypes, Problem-solving.
Description: Girls prefer gender-oriented play. Using women models in games triggers greater interest among girls in technical subjects [2].
Activity: Promote serious games with women protagonists and women role models in games to teach school subjects.

Integrated Curriculum Project Using Project-Based Learning (PBL) Activities
Topic: STEAM Area Awareness, Computational Thinking, Problem-solving.
Description: Collaborative projects that integrate computer science with other topics effectively engage girls. The project-based learning approach allows students to see computer science's practical and multidisciplinary relevance. This encourages students to use digital skills while learning, increasing the motivation of students to STEAM area [19].
Activities:

1. Math and Coding: Using programming to solve maths problems.
2. Science Experiments: Using sensors and programming to collect and analyse data.
3. The ICT-integrated English lessons and activities are prepared and conducted to prepare the students for real-life experiences.

(continued)

4. Promote a collaborative STEAM activity event yearly. The school can ask teachers from different departments related to the STEAM area to prepare an event together. It can be planned as a competition [15].
5. Developing an app or website related to a personal or social theme, such as sustainability or health.

Game-Based Learning
Topic: Logical thinking, Problem-solving, Programming.
Description: Educational games that teach programming concepts fun and engagingly have been shown to be effective for girls [71].
Activities:

1. Solving programming puzzles in Lightbot.
2. Promote challenges using Code.org resources.

Involve the Parents
Topic: Parental support, Awareness, Encouragement.
Description: Engaging parents as advocates for computer science can be highly beneficial. By informing parents about the opportunities and importance of computing, and demonstrating how it can empower their children, we can encourage them to support their daughters' exploration of this field [45, 63].
Activity: The school can proactively engage girls and their parents through informative sessions and other activities. Including parents in activities, workshops, or information sessions related to computer science can inspire them to become active participants in their children's learning journey and encourage their daughters to consider Informatics a viable and exciting career path.

Collaborative Learning Environments
Topic: Programming, Problem-solving, Collaboration, Teamwork.
Description: Activities that promote collaboration and teamwork are particularly effective for girls, as they create a supportive and encouraging environment [7].

(continued)

Activities (Examples):

1. Pair Programming: Students work in pairs, taking turns writing code and providing feedback to each other.
2. Peer Feedback Activities: Students share their projects with classmates, providing constructive criticism and suggestions for improvement.

Inquiry-Based Activities
Topic: Problem-solving.
Description: Providing students with opportunities to engage in hands-on, inquiry-based activities that reflect real-world STEM applications is crucial for sparking interest [13].
Activities (Examples):

1. Create a Tourism Information System.
   - *Subtopic:* Programming, Information Systems, Time Management.
   - *Description:* Students developed a tourism information system using Micro:bit to help tourists plan their trips. They learned about coding, data display, and calculating travel times.
2. VR Panorama.
   - *Subtopic:* Virtual Reality, Tourism Promotion, Design Thinking.
   - *Description:* Students created VR panoramas using Google Cardboard to showcase Hong Kong tourist attractions. They explored VR technology and its application in tourism marketing.
3. Micro:bit Flood Warning System
   - *Subtopic:* Sensor Technology, Flood Warning Systems, Engineering Design.
   - *Description:* Students designed and built a flood warning system using Micro:bit or Mbot to simulate real-world applications. They learned about sensor technology, programming, and designing solutions for environmental challenges.

4.4.1 Coding with XLogo in Elementary Schools in Ticino

Getting a better understanding of how children perceive a software engineer is a great starting point when planning how to help them be more aware of implicit stereotypes and their influence of future job and study careers. Besides, when running the DASET study described above it emerged also that children did not really have a clear idea of what a software engineer did and this is a good argument to advocate for early intervention when it comes to introduce elements of Informatics to elementary school children. Here we describe an ongoing effort in Ticino, one of Switzerland cantons, where elementary school teachers, university professors and students of Informatics have come together to provide children a taste of coding. Started in 2022, the initiative has involved three schools and five different classes in primary 3, 4, and 5, engaging children in a 10/8 weeks long introduction to coding with XLogo, via an online platform and the related paper textbook [39]. The textbook is divided into 4 chapters that cover 3 essential topics of programming:

1. Writing codes to define and control computer's operations.
2. Using parameters, cycles, and conditional functions for the development and implementation of real programs.
3. Introduction to modular programming to solve problems in a structured and comprehensive method.

Each chapter has a brief theoretical explanation and practical examples of the new concepts and commands introduced, it offers numerous exercises for the reader to practice and explore. The exercises do not give out the code, just the expected visual output. This allows the reader to experience and explore the "Trial and Error"' method, very common in programming fields, that is considered fundamental in developing problem-solving skills. The result of each exercise consists of a drawing that can represent just a straight line, a polygon, a circle, or a combination of all of them that can reproduce a common object such as stairs, a country's flag, or a humanoid face. The textbook is using the programming environment XLogo which is based on the educational programming language Logo, which first appeared back in 1967 and is now at the core of many programming languages developed for children.

In our initiative, after consulting also with the school teacher, we decided to use interactive teaching methods to teach programming as a natural way to involve children and make the subject fun. Overall, the purpose of a command is to make a turtle move in a certain direction and in doing so tracing a line that would compose a drawing, simple as a square in the beginning to become more complex while acquiring more commands. Commands are quite intuitive with simple syntax for children to follow. Also, the textbook is introducing each command one by one and allows the users to get confident with one command before moving to the next one.

The teaching team was composed by university professors, PhD, Master and bachelor students according to availability so that plenty of technical support was available when needed.

Children were working in pairs, organised by their teachers, sharing one laptop, and consulting their copy of the textbook. Each of the chapters of the textbook was divided in two parts to be administered separately, and each lesson would start with a recap. The teacher would act as facilitator and made sure children were comfortable and stayed focused. Each lesson lasted two teaching units, of 45 plus 45 minutes, with children too engrossed in the activity to give up their afternoon break and ask for exercises to take home. One child also asked her parents to move her regular doctor appointment to a different slot not to miss the lesson.

By gathering feedback from children and teachers, via interviews, open discussion, written assessments and direct observations, during this three years, we could appreciate how introducing coding via Xlogo in a hand-on fashion, provided a fun and engaging activity. Having the possibility to work in a classroom weekly for 8–10 weeks, let us observe changes in children's attitude but even after the novelty effect waived off, enthusiasm was always very high. Even reluctant participants got involved and enjoyed the experience of learning by doing. Girls were a bit more cautious in the beginning as less used to the tinkering paradigm adopted naturally by boys, but with a little extra encouragement, soon became very keen and proficient. Boys were more likely to go for the fun and even if in each class we made sure the textbook was read aloud and each new command explained by the tutor, boys liked best to experiment on their own, proud to share their drawings specially when different than the one mentioned in the textbooks. Quite on the opposite, girls would take pride when achieving exactly what the textbook exercise required, and would be careful to follow the sequence of activities proposed in the class. Teachers expressed their satisfaction too, with one commenting on how one of the"intellectual girls" in the classroom, spending her spare time during school break reading books, got captured by the simple exercises and enjoyed them too. Another teacher remarked on how a girl in her class had made great progress in mathematics once she got enticed with Xlogo. We also noticed how independent children grew in using both textbook and the programming environment as soon after the second meeting instead of asking for help, when raising their hands they were showing off their achievements. Our hope is that when growing up these children, and in particular the girls in these classes, will keep fond memories of their first encounter with Informatics, by associating it with a rewarding challenge they proved able to take on and achieve.

4.5 After School Activities

After-school computer science activities, ranging from elementary to high school, play an important role in addressing imbalance in Informatics because these activities provide early exposure, foster interest, and build confidence among girl students, which are essential for increasing gender diversity in computer science. According to Mouza et al. [54], CS knowledge is increasing with the after-school programs because students do not see it as assignments but practice it for fun. Early exposure to Informatics concepts is crucial for developing sustained interest and confidence in the subject, particularly among girls. Research has shown that integrating playful learning activities in after-school programs can effectively engage young students [23, 57, 72]. No One Left Behind (NOLB) Project is developed to involve young girls in programming and project results show that after-school coding courses had a significant impact on girl students' engagement and made them feel comfortable towards computer science [66]. Such projects can inspire students to engage more with computer science. Therefore, collaborations with the initiatives and including more projects are needed to encourage young girls to be in the computer science field just like how boys are doing. **After-school programs in elementary schools** are suggested as creating a playful learning environment, bringing role models in class to create a communication area for students and engaging competitions in the activities [47, 60, 62]. The following suggestions show how to include Informatics in after-school programs for the elementary school level to help gender balance in Informatics.

1. **Playful Learning Environment**: Game-based learning can make computer science fun and engaging for young students. Integrating educational games that teach Informatics to high school students boosts learning motivation and improves students' problem-solving, programming and critical-thinking skills [43]. Such activities make students more comfortable towards Informatics. For instance, Code.org activities can be integrated in after-school activities which offer a variety of coding games that incorporate popular themes and characters from movies and cartoons [1, 44, 46].
2. **Role Models in Class**: Having women role models in the classroom who have careers in Informatics can inspire young girls and show them that they belong in the field. Providing access to women role models and mentors in after-school activities can have a valuable impact on young girls' perception of computer science because the presence of women who are successful in the field can inspire and motivate students. According to Chen et al. [18], role models have an impact on students' career choices, so bringing women role models into the classroom at an early age can help remove the stereotype that Informatics is a men-dominated field. Cooperation with organizations or initiatives can be included in after-school activities to bring in women engineers and computer scientists to share their experiences and mentor students.
3. **Competitions**: Creating inclusive and supportive environments, these programs help to normalize the idea that Informatics is for everyone, regardless of gender. Therefore, competitions with the level of elementary schools can be included in the programs. Jr FLL: Junior FIRST LEGO League can be held to inspire students. FIRST® LEGO® League introduces children ages 4–16 to science, technology, engineering and maths (STEM) in an engaging and fun way that promotes hands-on learning experiences [55]. Through a dynamic global robotics programme, participants develop real-world problem-solving skills, helping students and educators work together for a better future. These competitions require students to work in teams to design, build, and program robots to complete specific tasks [70]. Research has shown that participation in robotics competitions can positively impact girls' perceptions of their abilities in STEM fields [55, 70].

After-school programs in high schools is important for shifting from simple, visual programming to text-based programming with written algorithms. At this stage, students begin to learn problem-solving in maths and science classes more seriously than in elementary school. After-school Informatics related activities offer several advantages over traditional classroom instruction. These programs are often more flexible, allowing for creative and hands-on learning experiences that can be tailored to the interests and needs of the students. Additionally, after-school programs can provide a more inclusive and supportive environment, which is particularly important for girls who may feel marginalized or discouraged in a men-dominated classroom [57]. A strategy should be developed in middle and high schools' after-school programs to promote gender diversity because by providing an engaging environment, these programs can encourage more girls to pursue Informatics, thereby helping to close the gender gap in this critical field [59, 64]. The following suggestions show how to include computer science in after-school programs for the elementary school level to help gender balance in Informatics.

1. **Hands-On Projects and Competitions**: Organizing coding clubs and hackathons that focus on collaborative projects rather than individual competition can raise the eagerness of students towards Informatics. This can reduce the intimidation factor and encourage participation from all students. Project-based learning (PBL) is an educational approach that involves students in complex, real-world projects through which they develop and apply skills and knowledge. PBL can be integrated into after-school programs to engage students because it increases students' motivation as a hands-on project [42]. Students can prototype an application that addresses real-world problems and works on the topic they are passionate about. Also, robotics activities can be included as an exciting way for students to apply their programming skills. Robotics projects often involve teamwork and problem-solving, which can be particularly appealing and empowering for female students[50]. Different competitions can be involved such as FLL: FIRST LEGO League can be held Bebras Challenges are organised. FLL is an international robotics competition that addresses students from 9 to 16 years old which takes place in lots of countries [27]. Usart et al. [69] indicates FLL increases students' problem solving skills which enables them to have twenty-first century skills naturally. On the other hand, competitions like Bebras, which is an international challenge in Informatics running for elementary and middle school students, increases students' motivations towards computer science [24].

After-school programs in high schools is important because despite ongoing efforts to address this issue, the disparity remains, particularly in high school education where foundational interests and skills are developed. To increase the awareness about Informatics for all genders, after-school programs present a valuable opportunity to engage girl students, fostering a diverse and inclusive environment that can lead to greater gender equity in the tech industry [32, 38]. To understand why there are fewer girl students in Informatics, researchers mainly look at high school education. They found that girls often feel uncomfortable with

Informatics in high school due to limited access to programming, a men-focused curriculum with abstract programming concepts, lack of real-world applications, and both subtle and obvious stereotypes from school staff [28]. The activities listed below are considered the best ways to incorporate Informatics into the high school after-school programs. The aim is to show that Informatics is more than just computers or coding by integrating it into all subjects, which can help change the "geek" stereotype of men role models for girl students.

1. **Providing Mentorship and Role Models**: Mentorship programs can play a significant role in encouraging girls to pursue computer science. Women mentors and role models can inspire students by sharing their experiences and providing guidance [22]. After-school programs can facilitate mentorship through guest speakers, workshops, and one-on-one mentoring sessions.
2. **Real-World Opportunities**: Connecting Informatics concepts to real-world applications can make the subject more appealing to female students [3]. Schools can collaborate with the initiatives to have projects after-schools with the students where students can work with a real client to see how Informatics is in the real world. Also, after-school programs can include activities such as field trips to tech companies, virtual tours, and guest lectures from industry professionals [14, 20]. These experiences can help students see the relevance of Informatics in various fields and inspire them to explore further.
3. **Promoting the Competition**: Effective promotion of competitions is vital in attracting students to after-school computer science programs [50]. This can be achieved through school announcements, flyers, social media, and events. Highlighting the benefits of participation, such as skill development, college preparation, and career opportunities, can motivate students and their families to get involved. Student teams for FRC: FIRST Robotics Competitions and WRO: The World Robot Olympiad can be formed to encourage more students. Also, science fair, hackathons per term can be organized to facilitate increased internet usage and technologies. Families can be involved during the process of competitions because involving parents and community members, such as coding nights, exhibitions, and competitions can raise awareness about the importance of gender diversity in Informatics and garner support for students' participation [65].

Promoting gender diversity in Informatics through after-school activities from elementary to high school is a multifaceted approach that requires commitment, collaboration, and creativity. Schools should work not only with teachers but with initiatives, parents and local committed to raise more awareness about the gender balance issue in computer science. By creating inclusive curricula, providing mentorship, fostering hands-on learning, addressing stereotypes, and involving families and communities, K-12 schools can create a supportive environment that encourages girl students to pursue Informatics. Successful implementation of these programs can lead to a more diverse and innovative tech industry, benefiting society as a whole.

4.6 Open Questions

In this chapter we started from elaborating on children's perception of Informatics as a profession and highlighted the presence of stereotypes at an early age. We then build on those insights, that were providing us with a better understanding of children's mental models, to ground a strong motivation for supporting the introduction of activities to engage girls with Informatics at school and after-school. Open questions stay on how to measure the effects of these activities and their effectiveness not only in terms of career choices but perhaps more importantly towards the overall objective of fighting against the obstacles that girls and women away from Informatics. For that we need to further explore the role played by the culture and society that are heavily influencing career choices of both boys and girls. Therefore, open questions stay on how families and educators as well as policy makers could better support and encourage girls in their choices. Next chapters will further elaborate on these open questions and add more, while providing evidences and pointing to relevant research directions.

References

1. Eunice Agyei. How to design activities for learning computational thinking in the context of early primary school in an after-school code club. Master's thesis, E. Agyei, 2019.
2. S. AlSulaiman and M. S. Horn. Peter the fashionista? computer programming games and gender-oriented cultural forms. In *CHI PLAY 2015 - Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play*, pages 185–196. ACM, 2015.
3. Rukiye Altin and Andreas Mühling. Why female students are dropping out of cs programs. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*, pages 304–310. 2024.
4. F. K. Bailie. Women who make a difference: Role models for the 21st century. *ACM Inroads*, 6(2):63–67, 2011.
5. J. Black, P. Curzon, C. Myketiak, and P. W. McOwan. A study in engaging female students in computer science using role models. In *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education*, pages 63–67. ACM, 2011.
6. Sandra Lipsitz Bem. Gender schema theory and its implications for child development: Raising gender-aschematic children in a gender-schematic society. *Signs*, 8(4):598–616, 1983.
7. P. S. Buffum, M. Frankosky, K. E. Boyer, E. N. Wiebe, B. W. Mott, and J. C. Lester. Collaboration and gender equity in game-based learning for middle school computer science. *Computing in Science & Engineering*, 18(2):18–28, 2016.
8. Anne-Jorunn Berg and Merete Lie. Feminism and constructivism: Do artifacts have gender? *Science, Technology, & Human Values*, 20(3):332–351, 1995.
9. Robertta H Barba and Cheryl L Mason. The emergence of the "nerd" an assessment of children's attitudes toward computer technologies. *Journal of Research on Computing in Education*, 26(3):382–390, 1994.
10. Valeria Borsotti. Barriers to gender diversity in software development education: actionable insights from a danish case study. In *Proceedings of the 40th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, pages 146–152. ACM/IEEE, 2018.

11. Mark Jeremy Brosnan. A new methodology, an old story? gender differences in the "draw-a-computer-user" test. *European Journal of Psychology of Education*, 14(3):375–385, 1999.
12. Philipp Brauner, Martina Ziefle, Ulrik Schroeder, Thiemo Leonhardt, Nadine Bergner, and Birgit Ziegler. Gender influences on school students' mental models of computer science: a quantitative rich picture analysis with sixth graders. In *Proceedings of the 4th Conference on Gender & IT*, pages 113–122, 2018.
13. Yu Chen, Stephen Cheuk Fai Chow, and Winnie Wing Mui So. School-stem professional collaboration to diversify stereotypes and increase interest in stem careers among primary school students. *Asia Pacific Journal of Education*, 42(3):556–573, 2022.
14. Sapna Cheryan, Benjamin J Drury, and Marissa Vichayapai. Enduring influence of stereotypical computer science role models on women's academic aspirations. *Psychology of women quarterly*, 37(1):72–79, 2013.
15. J. Century, K.A. Ferris, and H. Zuo. Finding time for computer science in the elementary school day: a quasi-experimental study of a transdisciplinary problem-based learning approach. *IJ STEM Ed*, 7(20):–, 2020.
16. Ching-Ching Cheng and Kuo-Hung Huang. Stereotypes and technology education: Different perceptions of computer career among elementary school students. *Journal of Baltic Science Education*, 15(3):271–283, 2016.
17. David Wade Chambers. Stereotypic images of the scientist: The draw-a-scientist test. *Science education*, 67(2):255–265, 1983.
18. Chen Chen, Jonathan Rothwell, and Pedrito Maynard-Zhang. In-school and/or out-of-school computer science learning influence on cs career interests, mediated by having role-models. *Computer Science Education*, pages 1–25, 2023.
19. Jord. Clarke, Deborah Silvis, Jessica F. Shumway, Victor R. Lee, and Joseph S. Kozlowski. *Assessing Computational Thinking*, chapter Developing a kindergarten computational thinking assessment using evidence-centered design: the case of algorithmic thinking. Routledge, 2023.
20. Sapna Cheryan, John Oliver Siy, Marissa Vichayapai, Benjamin J Drury, and Saenam Kim. Do female and male role models who embody stem stereotypes hinder women's anticipated success in stem? *Social psychological and personality science*, 2(6):656–664, 2011.
21. Claudia Maria Cutrupi, Irene Zanardi, Letizia Jaccheri, and Monica Landoni. Draw a software engineer test-an investigation into children's perceptions of software engineering profession. In *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*, pages 37–47. IEEE, 2023.
22. Wendy DuBow, Alexis Kaminsky, and Joanna Weidler-Lewis. Multiple factors converge to influence women's persistence in computing: A qualitative analysis. *Computing in Science & Engineering*, 19(3):30–39, 2017.
23. Jill Denner, Jacob Martinez, Julie Adams, and Heather Thiry. Computer science and fairness: Integrating a social justice perspective into an after school program. *Science Education and Civic Engagement*, 6(2):49–62, 2015.
24. Valentina Dagienė and Sue Sentance. It's computational thinking! bebras tasks in the curriculum. In *Informatics in Schools: Improvement of Informatics Knowledge and Perception: 9th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2016, Münster, Germany, October 13–15, 2016, Proceedings 9*, pages 28–39. Springer, 2016.
25. Joice J. Endendijk and Christel M. Portengen. Children's views about their future career and family involvement: Associations with children's gender schemas and parents' involvement in work and family roles. *Frontiers in Psychology*, 2022.
26. Organisation for Economic Co-operation and Development. *Why Do More Young Women Than Men Go on to Tertiary Education?* OECD Publishing, 2021.
27. https://www.firstinspires.org/. FIRST Lego League
28. Allan Fisher and Jane Margolis. Unlocking the clubhouse: the carnegie mellon experience. 34(2), 2002.

29. Yesharim Mor Friebroon and Ben-Ari Mordechai. Teaching computer science concepts through robotics to elementary school children. *International Journal of Computer Science Education in Schools*, 2(3):–, 2018.
30. Donna Farland-Smith. How does culture shape students' perceptions of scientists? cross-national comparative study of american and chinese elementary students. *Journal of Elementary Science Education*, 21(4):23–42, 2009.
31. Sarah Fox, Rachel Rose Ulgado, and Daniela Rosner. Hacking culture, not devices: Access and recognition in feminist hackerspaces. In *Proceedings of the 18th ACM conference on Computer supported cooperative work & social computing*, pages 56–68, 2015.
32. Joanna Goode. Increasing diversity in k-12 computer science: strategies from the field. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '08, page 362–366, New York, NY, USA, 2008. Association for Computing Machinery.
33. Linda S Gottfredson. Circumscription and compromise: A developmental theory of occupational aspirations. *Journal of Counseling psychology*, 28(6):545, 1981.
34. Linda S. Gottfredson. *Gottfredson's theory of circumscription, compromise, and self-creation*, chapter 4, pages 85–148. Jossey-Bass, 2002.
35. Alexandria K Hansen, Hilary A Dwyer, Ashley Iveland, Mia Talesfore, Lacy Wright, Danielle B Harlow, and Diana Franklin. Assessing children's understanding of the work of computer scientists: The draw-a-computer-scientist test. In *Proceedings of the 2017 ACM SIGCSE technical symposium on computer science education*, pages 279–284, 2017.
36. H. S. Hsiao, Y. W. Lin, K. Y. Lin, C. Y. Lin, J. H. Chen, and J. C. Chen. Using robot-based practices to develop an activity that incorporated the 6e model to improve elementary school students' learning performances. *Interactive Learning Environments*, 30(1):85–99, 2019.
37. John L Holland. *Making vocational choices: A theory of vocational personalities and work environments*. Psychological Assessment Resources, 1997.
38. Hoffman, B., Morelli, R. & Rosato, J. Student Engagement is Key to Broadening Participation in CS. *Proceedings Of The 50th ACM Technical Symposium On Computer Science Education*. pp. 1123–1129 (2019), https://doi.org/10.1145/3287324.3287438
39. Juraj Hromkovic. *Semplicemente Informatica*. Klett und Balmer AG, Baar, 2018.
40. Our World in Data. Number of people using the internet, 2022.
41. Sharin Rawhiya Jacob, Miranda C. Parker, and Mark Warschauer. *Integration of computational thinking into English language arts*, page 55–63. Association for Computing Machinery, New York, NY, USA, 2022.
42. Dimitra Kokotsaki, Victoria Menzies, and Andy Wiggins. Project-based learning: A review of the literature. *Improving schools*, 19(3):267–277, 2016.
43. Bolganay Kaldarova, Bakhytzhan Omarov, Lyazzat Zhaidakbayeva, Abay Tursynbayev, Gulbakhram Beissenova, Bolat Kurmanbayev, and Almas Anarbayev. Applying game-based learning to a primary school class in computer science terminology learning. In *Frontiers in Education*, volume 8, page 1100275. Frontiers Media SA, 2023.
44. Christopher Levy. *Elementary teacher professional development for computer science and digital game-based learning*. PhD thesis, Concordia University (Oregon), 2019.
45. Adam Lloyd, Jennifer Gore, Kathryn Holmes, Max Smith, and Leanne Fray. Parental influences on those seeking a career in stem: The primacy of gender. *International Journal of Gender, Science and Technology*, 10(2):308–328, 2018.
46. Dragan Lambić, Biljana orić, and Saša Ivakić. Investigating the effect of the use of code. org on younger elementary school students' attitudes towards programming. *Behaviour & Information Technology*, 40(16):1784–1795, 2021.
47. José Miguel Merino-Armero, José Antonio González-Calero, and Ramón Cózar-Gutiérrez. The effect of after-school extracurricular robotic classes on elementary students' computational thinking. *Interactive Learning Environments*, 31(6):3939–3950, 2023.
48. Emma M Mercier, Brigid Barron, and KM O'connor. Images of self and others as computer users: The role of gender and experience. *Journal of computer assisted learning*, 22(5):335–348, 2006.

49. Allison Master, Sapna Cheryan, and Andrew N Meltzoff. Computing whether she belongs: Stereotypes undermine girls' interest and sense of belonging in computer science. *Journal of educational psychology*, 108(3):424, 2016.
50. Allison Master, Sapna Cheryan, Adriana Moscatelli, and Andrew N Meltzoff. Programming experience promotes higher stem motivation among first-grade girls. *Journal of experimental child psychology*, 160:92–106, 2017.
51. M. Martínez, M. Gomez, and Benotti. A comparison of preschool and elementary school children learning computer science concepts through a multilanguage robot programming platform. In *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*, pages 159–164. ACM, 2015.
52. Florence F McCann, Edmund A Marek, et al. Achieving diversity in stem: The role of drawing-based instruments. *Creative Education*, 7(15):2293, 2016.
53. J. MWang and Hejazi Moghadam. Diversity barriers in k-12 computer science education: Structural and social. In *Proceedings of the 2017 ACM SIGCSE technical symposium on computer science education*, pages 615–620. ACM, 2017.
54. Chrystalla Mouza, Alison Marzocchi, Yi-Cheng Pan, and Lori Pollock. Development, implementation, and outcomes of an equitable computer science after-school program: Findings from middle-school students. *Journal of Research on Technology in Education*, 48(2):84–104, 2016.
55. Yuxin Ma and Douglas C Williams. The potential of a first lego league robotics program in teaching 21st century skills: An exploratory study. *Journal of Educational Technology Development and Exchange (JETDE)*, 6(2):2, 2013.
56. J. Noh and J. Lee. Effects of robotics programming on the computational thinking and creativity of elementary school students. *Educational Technology Research and Development*, 68:463–484, 2020.
57. Tom Neutens and Francis Wyffels. Bringing computer science education to secondary school: A teacher first approach. In *Proceedings of the 49th ACM technical symposium on computer science education*, pages 840–845, 2018.
58. Katarina Pantic, Jody Clarke-Midura, Frederick Poole, Jared Roller, and Vicki Allan. Drawing a computer scientist: stereotypical representations or lack of awareness? *Computer Science Education*, 28(3):232–254, 2018.
59. Viera K Proulx. Computer science in elementary and secondary schools. In *Informatics and Changes in Learning*, pages 95–101, 1993.
60. Kiki Prottsman. Computer science for the elementary classroom. *ACM Inroads*, 5(4):60–63, 2014.
61. Barbara J. Risman. *Gender as a Social Structure*, pages 19–43. Springer International Publishing, Cham, 2018.
62. Alpaslan Sahin, Mehmet C Ayar, and Tufan Adiguzel. Stem related after-school program activities and associated outcomes on student learning. *Educational Sciences: Theory and Practice*, 14(1):309–322, 2014.
63. Mara Šimunović and Toni Babarović. The role of parental socializing behaviors in two domains of student stem career interest. *Research in Science Education*, 51(4):1055–1071, 2021.
64. Sue Sentance, Mark Dorling, and Adam McNicol. Computer science in secondary schools in the uk: Ways to empower teachers. In *Informatics in Schools. Sustainable Informatics Education for Pupils of all Ages: 6th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2013, Oldenburg, Germany, February 26–March 2, 2013. Proceedings 6*, pages 15–30. Springer, 2013.
65. Linda J Sax, M Allison Kanny, Tiffani A Riggers-Piehl, Hannah Whang, and Laura N Paulson. "but i'm not good at math": The changing salience of mathematical self-concept in shaping women's and men's stem aspirations. *Research in Higher Education*, 56:813–842, 2015.
66. Bernadette Spieler and Wolfgang Slany. Female teenagers and coding: Create gender sensitive and creative learning environments. *arXiv preprint arXiv:1805.04366*, 2018.
67. Statista. Number of smartphones sold to end users worldwide from 2007 to 2023, 2024.

68. D. Scott, A. Zou, S. R. Jacob, D. Richardson, and M. Warschauer. Comparing boys' and girls' attitudes toward computer science. *Journal of Computer Science Integration*, 6(1):1–17, 2023.
69. Mireia Usart, Despoina Schina, Vanessa Esteve-Gonzalez, and Mercè Gisbert. Are 21st century skills evaluated in robotics competitions? the case of first lego league competition. In *CSEDU (1)*, pages 445–452, 2019.
70. Terri E Varnado. *The effects of a technological problem solving activity on FIRST™LEGO™League participants' problem solving style and performance*. PhD thesis, Virginia Polytechnic Institute and State University, 2005.
71. Maja Videnovik, Tone Vold, Linda Kiønig, Ana Madevska Bogdanova, and Vladimir Trajkovik. Game-based learning in computer science education: a scoping literature review. *IJ STEM Ed*, 10(54):–, 2023.
72. Timothy J Weston, Wendy M Dubow, and Alexis Kaminsky. Predicting women's persistence in computer science-and technology-related majors from high school to college. *ACM Transactions on Computing Education (TOCE)*, 20(1):1–16, 2019.
73. J. Wang, H. Hong, J. Ravitz, and Hejazi Moghadam. Landscape of k-12 computer science education in the us: Perceptions, access, and barriers. In *Proceedings of the 47th ACM technical symposium on computing science education*, pages 645–650. ACM, 2016.
74. Alexandra Wicht, Ai Miyamoto, and Clemens M Lechner. Are girls more ambitious than boys? vocational interests partly explain gender differences in occupational aspirations. *Journal of Career Development*, 49(3):551–568, 2022.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.