

Article

# Understanding Prospective Teachers' Task Design Considerations through the Lens of the Theory of Didactical Situations

Wajeeh Daher , Nimer Baya'a  and Otman Jaber

Mathematics Education Department, Faculty of Graduate Studies, Al-Qasemi Academic College of Education, Baqa 30100, Israel; bayaan@qsm.ac.il (N.B.); otmanj@qsm.ac.il (O.J.)

\* Correspondence: wdaher@qsm.ac.il

**Abstract:** Task design, in general, and task design in a technological environment, is attracting the attention of educational researchers. The present research investigates task design of prospective teachers in the Scratch programming environment. A total of twenty-three female prospective teachers participated in a professional development program. They were in their third academic year majoring in teaching mathematics and computer science in the middle school. The prospective teachers attempted to design mathematics-based programming problems. The present research utilizes the theory of didactical situations in mathematics, specifically the situation types, the paradoxes of the didactical contract and the situation components, to study the task design of the prospective teachers. It does that by focusing on one group of prospective teachers. The research results indicated that the prospective teachers were concerned mainly with the situation of information, situation of reference and situation of action. Doing so, they were concerned with the paradox of the said and the unsaid, the paradox of uncertainty, and the paradox of devolution. In addition, they took care of both algorithmic and creative reasoning. They also took care of students' devolution, where this devolution was conditioned with following an institutionalization. They were also concerned with giving students autonomy and encouraging decision making regarding the solution of the problem. Furthermore, they planned to enable students' control over their learning.

**Keywords:** task design; technological environment; scratch; didactical situation; prospective teachers; mathematics; programming



**Citation:** Daher, W.; Baya'a, N.; Jaber, O. Understanding Prospective Teachers' Task Design Considerations through the Lens of the Theory of Didactical Situations. *Mathematics* **2022**, *10*, 417. <https://doi.org/10.3390/math10030417>

Academic Editor: Jay Jahangiri

Received: 2 January 2022

Accepted: 26 January 2022

Published: 28 January 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Task design, in general, and mathematical task design, in particular, are attracting the attention of researchers as this design could help teachers plan, modify, build and evaluate activities in order to fit their classrooms [1]. This is especially true in times of educational emergencies as the last one that occurred due to COVID 19. The previous argument indicates that it is necessary for teachers' training colleges to prepare the prospective teachers for task design. The present research intends to verify the characteristics of task design by middle-school mathematics prospective teachers. The prospective teachers designed mathematics-based programming tasks in the Scratch environment. We will try to understand, in the present research, their task in light of the Theory of Didactical Situations in Mathematics (TDSM) and its components.

### 1.1. Programming in Mathematics Education

Forsström and Kaufmann ([2], p. 19) describe, based on Grover & Pea [3], how programming came to attract the attention of educators: "as programming has come to be recognized as a basic skill for effectively participating in the digital world, there has been increasing interest during the past decade in introducing programming as a school subject". Furthermore, Forsström and Kaufmann ([2], p. 19) argue that programming requires

having a knowledge of programming languages, in addition to “the ability to analyze, understand, and solve problems by verifying algorithmic requirements and assessing the correctness and implementation (often referred to as coding) of the algorithm in a particular programming language”. Grover and Pea [3] argue that the processes needed for programming have been linked to mathematical thinking, which makes it viable for mathematics learners.

### *1.2. The Scratch Programming Environment for Mathematics Education*

Scratch is defined by researchers as a visual programming language that is block-based, where learners use a sequence of code blocks to create a program. This block-based programming needs an effective pedagogy to make it an efficient tool for the consideration of mathematics in Scratch programming tasks [4,5]. Researchers reported that Scratch could encourage the cooperative learning of students [6] and encourage them to use problem-solving processes such as generating and testing of ideas and goal setting when learning mathematics [6]. Moreover, researchers found that Scratch can be used to develop both sixth-grade students’ computational thinking and mathematical ideas [7]. Calder [8] examined how children develop mathematical thinking when they work with Scratch. They reported that Scratch affected positively the affective aspects of children’s learning. These aspects included their motivation to explore mathematical concepts and their engagement in mathematical learning. Little research has been done on design in the Scratch environment. One attempt to do so was by Al-Dhlan and Al-Reshidi [9], who found that the design of E-games resulted in a statistically significant difference between the average scores of the research group, who used the designed E-games, in the technological awareness and the skill performance in visual programming in favour of the research group. Here, we attempt to verify how prospective mathematics teachers would design mathematics-based programming problems in the Scratch environment, when they participate in professional development program that involves solving and designing such problems.

### *1.3. Professional Development of Mathematics Teacher in Technology Integration in General and Programming Integration in Particular*

Technology integration, in the classroom in general and in the mathematics classroom in particular, has its advantages to students’ learning (e.g., [10,11]). These advantages include the cognitive aspect [12], the affective aspect [13], and the social aspect [14] as well as combinations of them [15]. These advantages of technology to students’ learning point at the need for teacher-training colleges to educate actual teachers and prospective teachers for technology integration in the classroom, so they will be prepared for effective integration of technology in their teaching. Researchers studied the education of actual teachers and prospective teachers in technology integration and how this education affected their teaching. Baya’a et al. [16] reported that the community of practice context can serve as an escalator that supports the decision of in-service mathematics teachers to integrate ICT in their mathematics teaching. The present research is interested in prospective teachers’ experiences while designing mathematics-based programming problems in the Scratch environment.

### *1.4. Task Design in Mathematics Education*

Cevikbas and Kaiser [17] say that task design constitutes a growing core of research in mathematics education. They say that the previous claim is particularly true for task design in dynamic and interactive mathematics learning environments. Tasks are described by researchers who follow the social constructivist framework as mediating tools for learning and teaching mathematics [11]. Cevikbas and Kaiser [17] say that different classification systems for mathematical tasks have been developed. Researchers point at some classifications for these tasks [18], where some examples of these classifications are: distinguishing investigations, problems, and exercises [19], or rich, authentic, and complex tasks [20].

Researchers talked about the relationship of task design and students' learning. Meslec et al. ([21], p. 3) relate task design to students' creativity: "When tasks differ in terms of content and types of cognitive processing, teams will experience the highest increase in creativity". Malicka [22] studied the impact of task design on students' language proficiency. She found that while high proficiency speakers took advantage of increases in cognitive complexity in terms of accuracy, low proficiency speakers did so at the level of structural complexity. In the present research, we consider the task design from the view of TDSM.

### 1.5. *The Theory of Didactical Situations in Mathematics*

Teaching situations can be described in terms of exchanges between students, teachers and the [23]. Students are interacting with other students, with the teacher and with the milieu. The milieu is defined as the whole things that influence the student or those that are influenced by the student: 'Within a situation of action, everything that acts on the student or that she acts on is called milieu' [23], (p. 9). Mackrell et al. [24], (p. 2655) stress the role of tasks in the didactical situation: "Key aspects of a didactical situation are the mathematical problem and the choice of didactical variable values to set for the task, where the task involves learning objectives and the mathematical problem. The teacher assumes that achieving the task will cause the student to learn". Thus, considering task design is needed to understand the role of the task on teaching and learning in the classroom.

In the present research, we will consider three characteristics of the didactical situation: the types of situation, the paradoxes in the didactical situation, and the components of the didactical situation. Below, we describe each characteristic.

#### 1.5.1. The Situation Types of the Didactical Situation:

Brousseau and Warfield [25] describe four types of didactical situations. Based on Brousseau and Warfield, the four types of didactical situations are: (1) Situation of reference. This occurs when a person, student, or teacher refers the person who asks to a piece of mathematical knowledge that belongs to their common repertoire; (2) situation of argumentation. This occurs when two persons engage in communication about an argument or a proof; (3) situation of information. This occurs when the two communicators cooperate on an action with success dependent on their common action; and (4) situation of action. This occurs when a subject intervenes in the situation to modify it with a specific aim.

#### 1.5.2. The Paradoxes of the Didactical Situation

Brousseau et al. [26] describe the paradoxes as part of the didactical contract and include seven types: (1) The paradox of custom and new knowledge. Here, the teacher can only attend to general procedures of a task, and the student cannot attend to a task of which she does not know the primary part. (2) The paradox of devolution. Here, the didactical contract can motivate the mathematics situation only by being broken. The student takes the responsibility, from which she releases the teacher, with all the risk entitled in it. (3) The paradox of the said and unsaid. Here, the student finds what she can say in what the teacher does not say. (4) The paradox of the actor. Here, the lesson is a show production, (5) The paradox of uncertainty. Here, mathematical situation cannot succeed without uncertainty or with too much uncertainty. (6) The paradox of adaptation of complex knowledge to too particular conditions. Here, extreme or early adaptation of complex knowledge to conditions that are too particular can result in it being replaced by a simplified and specific knowledge, and (7) the paradox of rhetoric and mathematics. Here, as well as acting as an educator who teaches the culture, including its historical mistakes, the teacher acts as an expert who causes rejection of parts disqualified by science.

#### 1.5.3. The Components of the Didactical Situation

Two components of TDSM are the devolution and institutionalization. "Devolution is the act by the teacher that makes the student accept the responsibility for an (adidactical)

learning situation or for a problem and accepts the consequences of this transfer of this responsibility.” ([23] p.230), while institutionalization occurs as “[the teacher] defines the relationships that can be allowed between the student’s ‘free’ behavior or production and the cultural or scientific knowledge and the didactical project; she provides a way of reading these activities and gives them a status.” ([23] p.56). Thus, it could be said that institutionalization occurs when arriving at the socially constructed mathematical knowledge [27].

Jonsson et al. [28] intended to allow for mathematical “struggle” in didactical situations (no teacher support) with tasks that are designed to facilitate students’ own construction of solutions. To do so, they suggested to consider Lithner [29] framework of creative and imitative reasoning. Doing so, they called the second type of reasoning algorithmic reasoning (AR). Artigue ([30], p. 160) says that we should pay attention to “the characteristics of the milieu with which the students will interact in order to maximize the potential it offers for autonomous action and productive feedback”. Thus, students’ autonomy is a factor that we need to consider when we come to design activities by following TDSM. Another factor is decision making of the student, as it affects the learning process.

### 1.6. Research Rationale and Goals

Cevikbas and Kaiser [17] say that task design research has been very popular, although it remains under-researched. They assert that overall, their results indicate that further studies are needed on task design in Dynamic and Interactive Mathematics Learning Environments. The Scratch environment is such an environment. The present research attempts to contribute to the task design research by investigating prospective teachers’ task design of mathematics-based programming problems in the Scratch environment. Little research has addressed this design issue, which points at the need of this research to shed light on the prospective teachers’ experimenting with task design.

Cevikbas and Kaiser ([17], p. 2) argue that “a strict classification of tasks is difficult to attain, as the appreciation of the complexity of a task is strongly connected to the student’s knowledge or competency level”. They conclude that teachers need to adjust classroom tasks according to the contexts of their classrooms so that they can implement these tasks effectively and achieve their goals.

The present research will utilize TDSM that describes three aspects of didactical situation to shed light on task design of mathematics-based programming problems. The first aspect is the situation type, while the second aspect is the paradoxes in the didactical situation. The third aspect is the components of the didactical situation. The prospective teachers’ work with task design could be understood by considering the three aspects of the didactical situation, which are detailed in the theoretical framework above. The present research attempts to utilize the previous three aspects to understand prospective teachers’ design of mathematics-based programming problems.

### 1.7. Research Questions

What are the types of didactical situations that prospective teachers consider while designing mathematics-based programming problems in the Scratch environment?

What are the paradoxes, in the didactical situations, that prospective teachers consider while designing mathematics-based programming problems in the Scratch environment?

How do the didactical situation components come into satisfaction in the prospective teachers’ design of mathematics-based programming problems in the Scratch environment?

## 2. Research Methodology

### 2.1. Research Design

The present research design is that of a case study. Case study is suitable in our case as the present research is pioneer in utilizing the didactical situation framework to analyze the task design of programming problems. We utilized two collecting tools to address the case study, which is the observation and the texts of the designed problems. This could

be considered a chain of evidence, meaning that each tool adds to the evidence on the different aspects of the didactical situation. Here, the observation supported the analysis of the didactical situation types and the paradoxes in the didactical situations, while the problem texts supported the analysis of the situation components.

## 2.2. Research Context and Participants

The context of the present research is that of a professional development program that was held for a full academic year (2020–2021). A total of twenty-three female prospective teachers aged 21–29, with mean age = 22.37 and SD = 2.29, participated in the professional development program. They were in their third academic year majoring in two disciplines for the middle school level: teaching mathematics and computer science. The prospective teachers' pedagogical supervisors were two of the authors. They accompanied the prospective teachers in their practical training in 2 middle schools. The prospective teachers worked in six groups of 3–4 prospective teachers. In the present research, we study task design by one of the groups in order to show this group's consideration of the three aspects of task design. We consider only one group's task design as this design is similar to the task design of the other groups in terms of the aspects in which the present research is interested.

At the beginning, the prospective teachers were engaged in mathematics-based programming problems as problem solvers, and afterwards as designers. In the present research we analyze the design of two problems by one of the groups, that of Lama, Amira and Abeer. The first problem involved the angles in a regular polygon, while the second was taken from (Becheanu [31], p. 305).

Let ABCD be a rectangular board with  $|AB| = 20$ ,  $|BC| = 12$ . The board is divided into  $20 \times 12$ -unit square squares. Let  $r$  be a given positive integer. A coin can be moved from one square to another if and only if the distance between the centers of the two squares is  $\sqrt{r}$ . The task is to find a sequence of moves taking the coin from square which has A as a vertex to the square which has B as a vertex.

Show that the task cannot be done if  $r$  is divisible by 2 or 3.

Prove that the task can be done if  $r = 73$ .

Can the task be done when  $r = 97$ ?

## 2.3. Research Collecting Tools

We requested the groups of prospective teachers to record their design sessions in Zoom. The group that we report here recorded their design discussions by voice only. When the groups handed us the recordings, we transcribed these recordings verbally.

## 2.4. Research Analysis Tools

We used deductive and inductive content analysis to analyze the prospective teachers' discussions of task design.

Below, Tables 1–3 include definitions and themes for the three aspects of the didactical situation that helped analyze the prospective teachers' design of mathematics-based programming problems.

## 2.5. Agreement between Coders as an Indicator of Reliability of the Data Analysis

To ensure the reliability between coders, we followed Syed and Nelson [32]. First, we developed a coding manual, which consisted of the codes, their descriptions and possible examples on them. Second, we tried to become familiar with the by carefully reading the transcription of the observation and the problem text. Third, we developed the categories and themes that emerged from the observation, and then from the problem text. The three authors coded the observations and the problem text, and the computation of agreement between coders, Cohen's Kappa, gave 0.91 in the case of the observation transcription and 0.95 in the case of the problem text. These results are suitable to conclude the reliability of the data analysis.

**Table 1.** Categories of didactical situation types, their definition, and their themes.

Situation Type	Definition	Themes
Reference	A situation in which there is reference to knowledge or experience of an expert	Refer to knowledge, refer to experience, refer to text
Argumentation	A situation in which there is reasoning, and justification of issues related to the situation	Reasoning concerning a mathematical claim, reasoning concerning the Scratch block suitable for a programming situation, reasoning about the Scratch block suitable for a mathematical situation.
Information	A situation in which there is communication about or cooperation regarding aspects of the situation	Communication about a problem, communication about a strategy, communication about a resource
Action	A situation in which there is intervening in order to modify an issue in the situation	Intervene, modify, change

**Table 2.** Categories of didactical situation paradoxes, their definition, and their themes.

Paradox Type	Definition	Themes
Custom	The paradox between the regular practices and the new content	Custom, regular, usual
Devolution	Giving the students control over their learning	Control, student’s role, learner’s role
The said and unsaid	What the teacher’s says and does not say	Teacher’s verbal saying, teacher’s expressions, teacher’s doing
Actor	Teaching as production and the classroom as stage	Acting, exploring, investigating
Uncertainty	Ambiguity as a result of the complexity of the situation	Little knowledge, complex situation, difficult problem,
Adaptation of complex knowledge	Widening complex knowledge	Adapt, adopt, adjust
Rhetoric and mathematics	Using metaphors to suit mathematical concepts	image, allegory, metaphor

**Table 3.** Categories of didactical situation components, their definition, and their themes.

Component in a Didactical Situation	Definition	Themes
Devolution	Accepting the responsibility of learning, solving, etc.	Responsibility for problem solving, responsibility for carrying out a project, responsibility for learning
Institutionalization	Arriving at the formal conception, general solution, etc.	Formal, scientific, precise
Algorithmic reasoning	Following a sequence of directions to solve a problem.	Solve according to method in the book, solve according to teacher’s method, solve according to an internet site
Creative reasoning	Thinking in a new or different way	Giving many solutions or ideas, giving different solutions or ideas, giving new solutions or ideas
Autonomy	The ability to choose how to work in a situation.	Choose a solving strategy, choose a learning method, choose to design one problem and not the other
Making decisions	Deciding upon an issue related to the situation	Make a decision, decide what strategy, decide what Scratch blocks to use

### 3. Results

We will describe here the prospective teachers’ design of mathematics-based programming problem in the Scratch environment. Presenting the prospective teachers’ design of

problems, we will describe the situation types that they were concerned with, the paradoxes that they attempted to avoid in the problems and the situation components that they discussed. We will describe one group's design to shed light on the design of mathematics-based programming problems by the prospective teachers. We will address the first and second research questions by considering the group's discussion during the design of the mathematics-based programming problems, while we will address the third question by considering the wording of the problem.

### 3.1. *The Group's Design of the First Mathematics-Based Programming Problem*

Coming to design the first mathematics-based programming problem, the group's members addressed different types of situation, in addition to four types of paradoxes: the paradox of devolution, the paradox of uncertainty, the paradox of the said and the unsaid and the paradox of custom. Below, we describe the situation types and the paradoxes by considering the discussion of the group's members regarding the problem.

#### 3.1.1. Situation of Action: Moving from the Previous Students' Knowledge of the Mathematics Situation to the New Knowledge

Lama, one of the group's members, started the discussion of the group by suggesting a mathematics-based programming problem (See transcript 1).

1. Lama: My idea depends on the regular polygon topic. The student, at the ninth grade, knows what a regular polygon is, what interior and exterior angles are, what rules there are for interior and exterior angles and what the number of the polygon diameters is. [5] My idea is that, with the help of the regular polygon's knowledge that I described before, the student draws the regular polygon.

**Transcript 1:** suggesting an action in the context of a mathematics-based programming problem.

In Transcript 1, Lama describes a situation of action of the problem suggested by her. Doing so, she describes her idea as concerned with moving from students' previous knowledge to the new in the context of a mathematics-based programming problem [R1]. Doing so, she starts the presentation of the situation of action with giving reasons why there is no paradox between the content of the mathematical problem and students' previous knowledge. It could be argued that Lama is concerned here with the paradox of devolution and the paradox of uncertainty. The students cannot learn if the uncertainty is too much for them to approach the problem, and in this case, they may not accept the devolution of the problem. Here, Lama tries to reason that this is the case of her suggested situation of action.

#### 3.1.2. Situation of Information: The Mathematics Content of the Problem and the Potentialities of the Problem

Amira discussed Lama's presentation of the situation of information by asking about the mathematical objects in her suggested programming problem.

2. Amira: Are you talking about writing a question about a specific polygon or a general polygon?
3. Lama: No, it is not necessary to write the questions about one polygon.
4. Abeer: O.K. I understand that.
5. Lama: If we make our questions about one polygon, the case will be very limited.
6. Abeer: I agree. If we talk about one polygon, the problem will not be interesting.
7. Lama: Yes, it will not lead to sufficient new thinking.

**Transcript 2:** the object of the mathematics-based programming problem.

In Transcript 2, Amira asked Lama about the situation of action associated with the problem. The question of Amira led the group members to discuss whether the mathematical objects of the situation of information were just one or different ones [R2–R4],

and how this might affect the potentiality of the problem to be not limited [R5], i.e., an interesting one [R6] or a problem that leads to sufficient new thinking [R7]. We can interpret Transcript 2 as referring to a situation of information. Here, the group's members are concerned with the potentiality of the information in the problem to give the students satisfactory new thinking. We argue that the group's members are also concerned with the paradox of the said and the unsaid, where here they wanted to make the 'said' less in order for the students to have they 'say' when they come to solve the problem.

This concern with the said and the unsaid is complemented by the paradox of uncertainty but here, and differing from the first transcript, they were concerned that the uncertainty was not too low.

### 3.1.3. Situation of Reference: The Mathematics and the Scratch References of the Problem

After describing the mathematical content in her main idea, Lama turned to talk about the references of her problem to mathematics content and Scratch content.

8. Amira: So?
9. Lama: So, this is my idea, to connect between Scratch and mathematics through regular polygons. I think this problem serves two goals. First, it serves to introduce the basic blocks in Scratch, the 'move' block, the 'turn' block, the 'repeat' block, etc. Second, the student consolidates her knowledge about regular polygons, especially knowledge about the interior and exterior angles in the regular polygon.
10. Amira: I notice a problematic issue in the situation that you describe, which is the new mathematical knowledge that the student gets here. There is almost no new mathematical knowledge.
11. Abeer: I agree with Amira. The student already knows everything about polygons, so we need to develop the problem in order to let the student benefit mathematically from solving the problem.
12. Lama: It was important for me that the problem includes both Scratch content and mathematics content.
13. Amira: O.K. Let me suggest another problem, and then we discuss again the situations in the problems.

**Transcript 3:** discussing the Scratch content and the mathematics content of the problem.

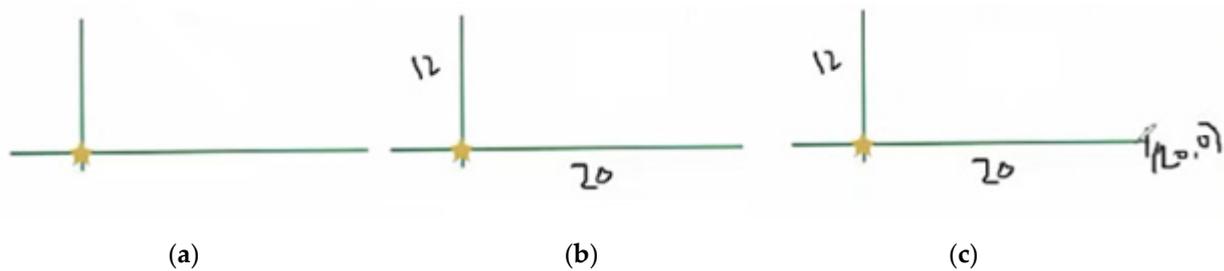
Transcript 3 involves a situation of reference. Here, the group's members discuss the reference of the suggested problem to the Scratch content and the mathematics content [R8–R13]. Lama explained that her goal, from the situation, was to refer to both contents, i.e., that the problem does not refer to one content only [12]. It could be said that the discussion of the group's members is concerned with fitting between the two requested references, the Scratch and the mathematical, i.e., that no reference comes at the expense of the other. In addition, it could be argued that the prospective teachers are concerned here with the custom paradox, where the custom is that a problem is only a mathematical problem or a programming problem, while the new knowledge was concerned with involving both types of knowledge, mathematical and programming. This led to Amira's suggestion to present her design of another problem.

### 3.2. The Group's Design of the Second Mathematics-Based Programming Problem Different Situation Types and Different Paradoxes

The group's members then discussed the second problem, paying attention to the same paradoxes to which they paid attention when discussing the first problem.

14. Amira: The problem I suggest is about a sprite that wants to move from a point into another point. The student is requested to move the sprite in different directions from a point into another point. The sprite moves only parallel to the  $x$ -axis or  $y$ -axis; upward, downward, to the right or left, in order to arrive at the end point.

15. Lama: Allow me to ask a question, so that I understand the situation. You are talking about the coordinates of points in the coordinate system. This is the mathematics knowledge that the student should possess. Right?
16. Abeer: Yes, this is what I also understood from Amira’s description.
17. Amira: Exactly, in addition to knowledge about the axes, the  $x$ -axis and the  $y$ -axis. We also want to limit the area in which the student works into the positive values of the  $x$ -axis and the  $y$ -axis, and  $x$  is limited to 20 while  $y$  is limited to 12. The student also needs to know how to factorize a number, say 73. She should know that it could be factorized into:  $3^2 + 8^2$ . So, the student needs to have previous knowledge about point coordinates and powers.
18. Lama: Let me understand the situation that you described. I tried to draw a drawing that depicts the situation [She draws the drawing in Figure 1a]. What did you say about the length of  $y$ ?  $y = 12$ ? O.K. and the length of  $x$ ,  $x = 20$  O.K. You said a sprite. I put a star [She draws as in Figure 1b]. Where do you want the sprite to get?



**Figure 1.** Point-to-point movement of a sprite in the positive X–Y plane. (a): draw a drawing that depicts the situation; (b): the sprite on (0,0); (c): the sprite move into (20, 0).

19. Amira: Last point on the  $x$ -axis.
20. Abeer: That means (20, 0).
21. Lama: [Lama draws as in Figure 1c. You mean that the sprite will move from (0,0) into (20, 0).
22. Amira: Exactly, but the idea is to give the student a condition on the distance between the starting point of the sprite to the end point. This condition is that the sprite makes two moves, generating a right-angle triangle, with hypotenuse equal to  $\text{Sqrt}(73)$ . This way, we ensure that the student does not move the sprite directly from (0,0) into (20,0).
23. Abeer: What is  $\text{Sqrt}(73)$ ?
24. Amira: The number of steps that the sprite in fact passes. The sprite can come back to a point that it already was at. We need to give the student three distances, where the first could satisfy the conditions of the problem, while the second could not. The third can satisfy or not satisfy the conditions of the problem.
25. Lama: I suggest giving the student another question, requesting her to choose a distance and finds out whether it satisfies the conditions of the problem.
26. Amira: O.K. Let us write the problem.

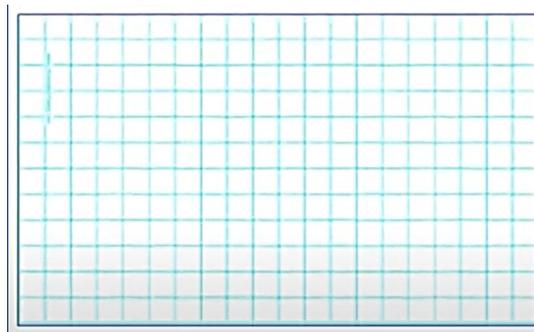
**Transcript 4:** Suggesting an alternative problem.

Transcript 4 represents a design of a mathematics-based programming problem that includes different types of situations [R15–R25]. Amira starts describing a situation of action-what the sprite needs to do [R14]. The group then discusses the situation of information related to the problem- the givens and the requirements of the problem as information [R15–R17]. It could be argued that Lama and Abeer try to understand the situation of information of the problem to guarantee that it does not include paradoxes. Together, they verify the givens and the requirements of the information situation [R15–R17], that it does not include either the paradox of uncertainty or the paradox of devolution. Moreover, they verify that the problem is not very simple [R22], that is it does not include a paradox of uncertainty. In addition, they verify that the problem covers the different possible solutions

of the problem [R24], again that it does not include the paradox of devolution that leads to the institutionalization situation, i.e., formal knowledge that includes all the accepted relations.

### 3.3. The Wording of the Mathematics-Based Programming Problem

You are requested to solve the following problem in Scratch environment, depending on your previous knowledge (Figure 2).



**Figure 2.** Scratch environment problem requiring sprite to move from the lower left end of the paper to the lower right.

Help the sprite to move from the lower left end of the paper to the lower right end of the paper ( $20 \times 12$ ).

In solving the problem, you should keep in mind that the sprite can move only parallel to the  $x$ -axis or  $y$ -axis; upward, downward, to the right or left, in order to arrive at the end point. In addition, the distance between two points could be described in terms of the sum of the squares of two numbers. To arrive from one to the next end of the paper, the sprite should move each time from one point to another on condition that it uses the combination of two allowed moves, and the distance between the two points is  $\text{Sqrt}(73)$  units.

#### Analysis of the Problem

Analysing the previous formulation of the problem in terms of the didactical situation components, the problem could be described as encouraging both algorithmic and creative processes. The algorithmic processes are indicated in giving the students a hint that the square of the distance can be described in terms of the sum of two quadratic numbers, while the creative processes are indicated in giving the students the option to let the sprite move in different paths on condition that the path's length is  $\text{Sqrt}(73)$ .

Moreover, some devolution was given for the students, as they were enabled to work on their own to get from one end of a rectangle into the other on condition that they move  $\text{Sqrt}(73)$  units. This devolution was conditioned with following an institutionalization about the movement between two points. The students have autonomy and could take decisions regarding the paths on which the sprite moves.

## 4. Discussion

The present study intended to investigate the prospective teachers' design of mathematics-based programming problems in terms of three characteristics of the didactical situation: the type of situation, paradoxes within the mathematics situation and the components of the mathematics situation. Below, we discuss the design according to each component.

### 4.1. Design According to the Type of Situation

The group that we studied took care primarily of the situation of action, the situation of information and the situation of reference. The situation of action served to describe how to move from the previous to the new mathematical knowledge in the context of the mathematics-based programming problem. In addition, it served to describe the action of

the sprite in the programming environment. The first occurrence of the action situation was interested with the success of the student to arrive at the new knowledge. The second occurrence of the situation of action was interested with the success of the sprite to describe the mathematical content. Both occurrences addressed the success of action in a specific didactical situation [31].

The group's discussion of the situation of information served to investigate how the mathematics content of the problem influence the potentialities of the problem. It could be said that the group wanted to make sure the students engage in problem solving in terms of Albay [33]: "the term 'problem solving' refers to mathematical tasks that have the potential to provide intellectual challenges for enhancing students' mathematical understanding and development". In addition, the discussion of the situation of information served to understand the givens the problem in order to guarantee that situation of information does not include paradoxes, especially the devolution paradox, where the solution of this paradox is that the problem lies in the Zone of Proximal Development' of the school students [34].

The group's discussion of the situation of reference served to verify that the mathematics-based programming problem includes both the mathematics reference and the programming reference. This situation of reference was essential as the problem had to include both references in order to suit the mathematics classroom and the programming environment of the problems. Shahbari et al. [35] found that the participating mathematics prospective teachers developed their meta-cognitive functions as problem solvers in both mathematics and programming aspects. Here, the prospective teachers made successful addressing of the situation of reference that considered both contents, the mathematical and the programming ones.

#### 4.2. Design According to the Paradoxes

The paradoxes, with which the prospective teachers were concerned, were the paradox of the said and the unsaid, the paradox of uncertainty, and the paradox of devolution. The previous paradoxes were presented in terms of students' knowledge, the potentialities of the problem and the mathematics and programming knowledges. First, those between students' knowledge and the mathematics situation. Second, those between the mathematics content of the problem and the potentialities of the problem, and third those between the mathematics concepts of the problem and the Scratch concepts of the problem. The first paradox is between the previous knowledge and the new knowledge. Researchers pointed at the need to connect between the previous knowledge and the new knowledge. Diaz [36] assert this position of the researchers: "Students vary differently. They differ in the places where they come from and along their journey from places to places to the classroom, they imbibed a broad-range pre-existing knowledge, beliefs, skills, attitudes which could affect how they receive, understand, and organize new knowledge. These broad preexisting are the prior knowledges of the learners. Prior knowledge may help or hinder the student in learning, depending on the nature of prior knowledge". The prospective teachers considered this connection in their design by making sure that there is no paradox between students' knowledge and the mathematics situation.

The second paradox that the prospective teachers considered is that between the mathematics content of the problem and the potentialities of the problem. This paradox considers the complexity of the problem, where the prospective teachers were concerned that their design of the problem turns the problem into an interesting one that leads to sufficient new thinking. It is suggested that the design of mathematics and science tasks should lie in the Zone of Proximal Development, i.e., "the space between what a learner can do without assistance and what a learner can do with adult guidance or in collaboration with more capable peers" [37]. Being in this zone ensures that the task will be considered interesting by the students, which indicates the important role of Zone of Proximal Development in task design.

The third paradox that the prospective teachers considered is that between the mathematics concepts of the problem and the Scratch concepts of the problem. It is expected that the prospective teachers pay attention to this paradox as the problems that they designed combined between the mathematics knowledge and the programming knowledge. This consideration of the fit between the mathematics knowledge and the programming knowledge would enable the students to see the link between the two [38], and thus utilize this link for students' learning of mathematics and programming.

#### 4.3. Design According to the Components of the Mathematics Situation

The research results indicated that the prospective teachers' design of mathematics-based programming problems took care of both algorithmic and creative reasoning. It also took care of students' devolution, where this devolution was conditioned with following an institutionalization about the distance between two points. The design of devolution was accompanied with giving students autonomy and decision making regarding the solution of the problem. The research results indicate that design in the Scratch environment enables students' control over their learning, as in other technological environments [39,40].

### 5. Limitations and Conclusions

One limitation of the present research is that it considered the design issue of activities in the Scratch environment in one of its phases, before the implementation. Future studies are needed to study this issue in the rest of the design phases, especially the influence of design on the implementation of the task and the influence of the implementation on the re-design of the task for future classroom practice. A recent study of Daher et al. [41] studied task design of science and mathematics teachers in emergency education, but did not address this influence on the redesign of the task for future classroom practice. The re-design issue needs special attention of educational researchers.

The research results indicate that the prospective teachers did not take care of the situation of argumentation in designing the mathematics-based programming problem. These results point at the need to make the prospective teachers aware of the situation of argumentation as a situation needed in mathematical task design. Researchers point at the problematic issue of argumentation in the mathematics classroom and call for paying attention to this issue. Yackel and Hanna [42] call "to design means to support teachers in developing forms of classroom mathematics practice that foster mathematics as reasoning and that can be carried out successfully on large scale". Castro et al. [43] say that "Teachers and students alike face similar difficulties in dealing with argumentation. One of the most challenging goals for mathematics teachers refers to helping students in the development of argumentation", which points at the need to make inservice and prospective teachers aware of the importance of argumentation in the mathematics classroom. This awareness would make them aware of this issue when they come to design activities for students learning.

Dohn [44] says that "All the referenced studies show that Scratch is a useful tool for coding in school settings and that its application has a positive effect on students' learning". The research results indicate that task design in the Scratch environment enables the devolution and institutionalization of students' learning, which points at the potentialities of Scratch environment to be an environment for students' voice. It is recommended that mathematics and programming teachers utilize this environment for designing activities that give students devolution over their learning.

**Author Contributions:** Conceptualization, W.D., N.B. and O.J.; methodology, W.D., N.B. and O.J.; software, N.B. and O.J.; formal analysis, W.D.; investigation, N.B. and O.J.; resources, N.B. and O.J. data curation, N.B. and O.J.; writing—original draft preparation, W.D.; writing—review and editing, W.D., N.B. and O.J. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Informed consent was obtained from all subjects involved in the study.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Daher, W.; Shahbari, J.A. Design of STEM activities: Experiences and perceptions of prospective secondary school teachers. *Int. J. Emerg. Technol. Learn. (IJET)* **2020**, *15*, 112–128. [\[CrossRef\]](#)
- Forsström, S.E.; Kaufmann, O.T. A literature review exploring the use of programming in mathematics education. *Int. J. Learn. Teach. Educ. Res.* **2018**, *17*, 18–32. [\[CrossRef\]](#)
- Grover, S.; Pea, R. Computational thinking in K–12: A review of the state of the field. *Educ. Res.* **2013**, *42*, 38–43. [\[CrossRef\]](#)
- Grover, S.; Pea, R.; Cooper, S. Designing for deeper learning in a blended computer science course for middle school students. *Comput. Sci. Educ.* **2015**, *25*, 199–237. [\[CrossRef\]](#)
- Han, B.; Bae, Y.; Park, J. The effect of mathematics achievement variables on Scratch programming activities of elementary school students. *Int. J. Softw. Eng. Appl.* **2016**, *10*, 21–30. [\[CrossRef\]](#)
- Taylor, M.; Harlow, A.; Forret, M. Using a computer programming environment and an interactive whiteboard to investigate some mathematical thinking. *Procedia Soc. Behav. Sci.* **2010**, *8*, 561–570. [\[CrossRef\]](#)
- Rodríguez-Martínez, J.A.; González-Calero, J.A.; Sáez-López, J.M. Computational thinking and mathematics using Scratch: An experiment with sixth-grade students. *Interact. Learn. Environ.* **2020**, *28*, 316–327. [\[CrossRef\]](#)
- Calder, N. Using Scratch: An integrated problem-solving approach to mathematical thinking. *Aust. Prim. Math. Classr.* **2010**, *15*, 9–14.
- Al-Dhlan, K.A.; Al-reshidi, H.A. Using Scratch Program to Design Educational E-Games to raise Technological Awareness and Visual Programming Skills to Face Fatal E-Games. *Indian J. Sci. Technol.* **2021**, *14*, 1364–1378. [\[CrossRef\]](#)
- Daher, W.; Sleem, H. Middle School Students' Learning of Social Studies in the Video and 360-Degree Videos Contexts. *IEEE Access* **2021**, *9*, 78774–78783. [\[CrossRef\]](#)
- Daher, W.; Baya'a, N.; Anabousy, R. In-Service Mathematics Teachers' Integration of ICT as Innovative Practice. *Int. J. Res. Educ. Sci.* **2018**, *4*, 534–543. [\[CrossRef\]](#)
- Swidan, O.; Daher, W. Low achieving students' realization of the notion of mathematical equality with an interactive technological artifacts. *EURASIA J. Math. Sci. Technol. Educ.* **2019**, *15*, em1690.
- Daher, W. Learning mathematics in the mobile phone environment: Students' emotions. *J. Interact. Learn. Res.* **2011**, *22*, 357–378.
- Daher, W.; Awawdeh Shahbari, J. Secondary Students' Identities in the Virtual Classroom. *Sustainability* **2020**, *12*, 4407. [\[CrossRef\]](#)
- Daher, W.; Swidan, O. Positioning–Emotions Association of Young Students Using Digital Technology. *Mathematics* **2021**, *9*, 1617. [\[CrossRef\]](#)
- Baya'a, N.; Daher, W.; Anabousy, A. The Development of In-Service Mathematics Teachers' Integration of ICT in a Community of Practice: Teaching-in-Context Theory. *Int. J. Emerg. Technol. Learn.* **2019**, *14*, 1. [\[CrossRef\]](#)
- Cevikbas, M.; Kaiser, G. A Systematic Review on Task Design in Dynamic and Interactive Mathematics Learning Environments (DIMLEs). *Mathematics* **2021**, *9*, 399. [\[CrossRef\]](#)
- Margolinas, C. *Task Design in Mathematics Education Proceedings of ICMI Study 22*; ICME: Oxford, UK, 2013.
- Joubert, M. Using computers in classroom mathematical tasks: Revisiting theory to develop recommendations for the design of tasks. In *Task Design in Mathematics Education. Proceedings of the ICMI Study 22, Oxford, UK, 22–26 July 2013*; Margolinas, C., Ed.; University of Oxford: Oxford, UK, 2013; pp. 69–77.
- Shimizu, Y.; Kaur, B.; Huang, R.; Clarke, D. The role of mathematical tasks in different cultures. In *Mathematical Tasks in Classrooms Around the World*; Brill Sense: Rotterdam, The Netherlands, 2010; pp. 1–14.
- Meslec, N.; Graff, D.; Clark, M.A. Increasing team ideation by sequencing the task type and content. *Des. Stud.* **2020**, *70*, 100947. [\[CrossRef\]](#)
- Malicka, A. *The Role of Task Complexity and Task Sequencing in L2 Monologic Oral Production*; University of Barcelona: Barcelona, Spain, 2014.
- Brousseau, G. *Theory of Didactical Situations in Mathematics*; Springer: Berlin/Heidelberg, Germany, 1998.
- Mackrell, K.; Maschietto, M.; Soury Lavergne, S. Theory of didactical situations and instrumental genesis for the design of a Cabri Elem book. In *Eighth Congress of European Research in Mathematics Education (CERME 8)*; Middle East Technical University: Ankara, Turkey, 2013; pp. 2654–2663.
- Brousseau, G.; Warfield, V. Didactic situations in mathematics education. *Encycl. Math. Educ.* **2020**, *3*, 206–213.
- Brousseau, G.; Sarrazy, B.; Novotná, J. Didactic contract in mathematics education. *Encycl. Math. Educ.* **2020**, 197–202. [\[CrossRef\]](#)
- Kislenko, K. Student's beliefs about mathematics from the perspective of the theory of didactical situations. In *Mathematicsthe French Way*; Winsløw, C., Ed.; Center for Naturfagenes Didaktik: København, Denmark, 2005; pp. 83–96.
- Jonsson, B.; Norqvist, M.; Liljekvist, Y.; Lithner, J. Learning mathematics through algorithmic and creative reasoning. *J. Math. Behav.* **2014**, *36*, 20–32. [\[CrossRef\]](#)
- Lithner, J. A research framework for creative and imitative reasoning. *Educ. Stud. Math.* **2008**, *67*, 255–276. [\[CrossRef\]](#)
- Artigue, M. Perspectives on design research: The case of didactical engineering. In *Approaches to Qualitative Research in Mathematics Education*; Springer: Dordrecht, The Netherlands, 2015; pp. 467–496.

31. Becheanu, M. *International Mathematical Olympiads: 1959–2000*; The Academic Distribution Center: Cary, NC, USA, 2001.
32. Syed, M.; Nelson, S.C. Guidelines for establishing reliability when coding narrative data. *Emerg. Adulthood* **2015**, *3*, 375–387. [[CrossRef](#)]
33. Albay, E.M. Analyzing the effects of the problem solving approach to the performance and attitude of first year university students. *Soc. Sci. Humanit. Open* **2019**, *1*, 100006. [[CrossRef](#)]
34. Little, D. Constructing a theory of learner autonomy: Some steps along the way. *Future Perspect. Foreign Lang. Educ.* **2004**, *101*, 15–25.
35. Shahbari, J.A.; Daher, W.; Baya'a, N.; Jaber, O. Prospective Teachers' Development of Meta-Cognitive Functions in Solving Mathematical-Based Programming Problems with Scratch. *Symmetry* **2020**, *12*, 1569. [[CrossRef](#)]
36. Diaz, K.V.L.T. *Prior Knowledge: Its Role in Learning*; University of the Philippines Los Baños: Los Baños, Philippines, 2017.
37. Billings, E.; Walqui, A. The Zone of Proximal Development: An Affirmative Perspective in Teaching ELLs/MLLs. 2018, p. 6. Available online: <https://www.wested.org/resources/zone-of-proximal-development/> (accessed on 1 January 2022).
38. Kilhamn, C.; Bråting, K.; Rolandsson, L. Teachers' arguments for including programming in mathematics education. In Proceedings of the NORMA20. Svensk förening för MatematikDidaktisk Forskning-SMDF, Oslo, Norway, 1–4 June 2021.
39. Daher, W. Student Voice in the Mobile Phone Environment: A Grounded Theory Approach. *Int. J. Mob. Blended Learn. (IJMBL)* **2017**, *9*, 12–23. [[CrossRef](#)]
40. Daher, W.; Baya'a, N. Characteristics of middle school students learning actions in outdoor mathematical activities with the cellular phone. *Teach. Math. Appl. Int. J. IMA* **2012**, *31*, 133–152. [[CrossRef](#)]
41. Daher, W.; Abo Mokh, A.; Shayeb, S.; Jaber, R.; Saqer, K.; Dawood, I.; Bsharat, M.; Rabbaa, M. The Design of Tasks to Suit Distance Learning in Emergency Education. *Sustainability* **2022**, *14*, 1070. [[CrossRef](#)]
42. Yackel, E.; Hanna, G. Reasoning and proof. In *A Research Companion to Principles and Standards for School Mathematics*; Kilpatrick, J., Martin, G., Schifter, D., Eds.; National Council of Teachers of Mathematics: Reston, VA, USA, 2003; pp. 22–44.
43. Castro, W.F.; Durango-Urrego, J.H.; Pino-Fan, L.R. Preservice Teachers' Argumentation and Some Relationships to Didactic-Mathematical Knowledge Features. *EURASIA J. Math. Sci. Technol. Educ.* **2021**, *17*, em2002. [[CrossRef](#)]
44. Dohn, N.B. Students' interest in Scratch coding in lower secondary mathematics. *Br. J. Educ. Technol.* **2020**, *51*, 71–83. [[CrossRef](#)]