Computational Thinking

A Learning Experience for Elementary Education (K6) Cycle

Initial 5 · First · Second Grade | Early Childhood and Primary Education

Pensamiento Computacional e Inteligencia Artificial

Computational Thinking

A Learning Experience for Elementary Education (K6) Cycle

Initial 5 · First · Second Grade | Early Childhood and Primary Education

Introduction

The computational thinking learning experience for the early school grades is designed to introduce computer science in a playful and structured way, providing strategies and learning opportunities for Kindergarten teachers, first, and second grade in early childhood and primary education.

Previous experiences from Ceibal in these grades are revisited; the Marco referencial de Pensamiento Computacional (Computational Thinking Referential Framework) (Ceibal, 2022), based on the experience of older elementary grades (4-6), as well as the main areas and guidelines of the Coding as Another Language Pedagogy and Curriculum, developed by Marina Umaschi Bers and the DevTech research group..

A variety of didactic sequences are proposed, incorporating coding and programming as key elements to foster developmentally appropriate experiences in expression, language, problem-solving, imagination, cognitive challenges, social interactions, motor and emotional skills, and decision-making (Bers, 2020). The implementation of this project aims to promote the development of cognitive and socio-emotional skills in children while serving as an expressive medium closely connected to emotions and social transformation.

Additionally, elements of digital literacy are included, which have progressively become essential competencies for this century. That is, the skills to use digital technology, communication tools, and networks to locate, assess, use, and create information (UNESCO, 2011) from a critical conceptual perspective on the technological environment. This supports individuals in becoming critical and active participants, rather than merely consumers of technology and digital content.

Activities focus on active students and take as their starting point the exploration of reality, the search for problems, their causes, their consequences, and the beliefs around them. This project aims to create opportunities to design and face challenges for the benefit of individuals.

Creativity, which expands horizons toward a rich and fulfilling life, is promoted through play, a vital element in the development and learning of children (Villarroel, 2012). The activity centers on play-based learning, which is essential for exercising freedom of choice and engaging in spontaneous activities that foster autonomy, independence, and confidence. This natural activity becomes a symbolic process of social communication (Lara Barrera, cited in Villarroel, 2012).

The didactic activities invite students to imagine what they want to do, create projects, play with their creations, share their ideas with the group, and reflect on their experiences, leading them to imagine new ideas and projects (Resnick, 2007).

Learning Goals

The proposed learning experience aims to develop skills related to problem-solving, algorithms, devices, and communication.

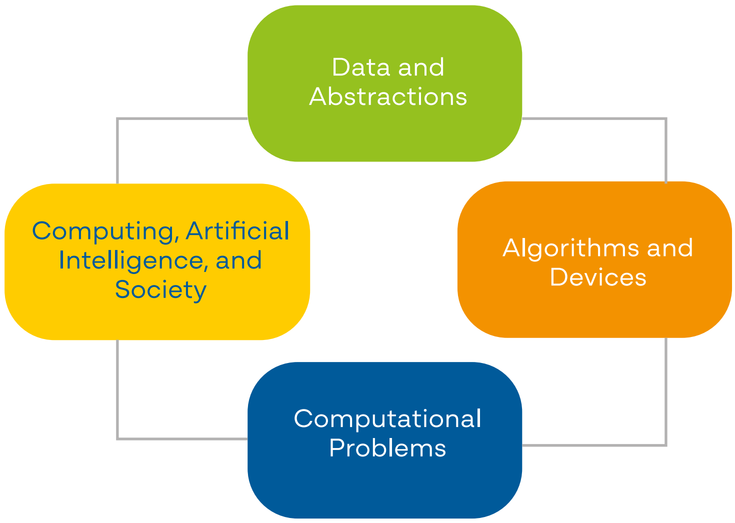
Through classroom implementation of these activities, students will be able to:

- Communicate ideas and emotions through animations, illustrations, and presentations using a block-based programming language.
- Develop basic notions about algorithms: their structure and their importance in solving simple problems.
- Connect knowledge and skills from different disciplines.

Ceibal's Referential Framework for CT

The general framework of the program is based on four dimensions and seven principles that guide the development of the didactic sequences.

Dimensions



Principles

- **Equity**
- **Collaboration**
- **Creativity**
- **Exploration**
- **Autonomy**
- **Critical Perspective**
- **Active Methodologies**

National Curriculum Framework and Computer Science and Educational Technology Program

This activity aligns with the computational thinking competency defined in the National Curriculum Framework (ANEP, 2022a), as well as with the Computer Science and Educational Technology programs within the Technical-Technological area (ANEP, 2023a; 2023b). Each activity details the specific competencies and contents addressed for each segment of these programs.

In a general sense, the following specific competencies of computational thinking (ANEP, 2022b) are addressed:

Computational Problem Solving

- Proposes solutions and tests them.
- Uses decomposition to break problems into simpler parts and identifies recurring patterns.

Data and Information Analysis

- Identifies symbols, such as images, that represent or have meaning.
- Establishes relationships between data or information in simple situations.
- Shares the data used in a resolution.
- Identifies useful and necessary data to solve problems and make simple predictions.

Algorithms and Devices

- Orders the events of a story in a logical sequence.
- Identifies the importance of order in algorithms.
- Identifies and proposes steps for solving simple problems and organizes them into a structured sequence.

- Creates and follows step-by-step instructions in playful or everyday situations.
- Uses programming as a form of expression.
- Identifies and combines basic programming commands.

Social Transformation

- Recognizes computers as integrated elements of everyday life.
- Describes cause-and-effect logic.

Pedagogical Activities

The learning experience for the early school years reaches the classroom through four diverse and complementary activities.

Supercircuit | Tell Me a Story | Step by Step | Up and Down

These activities gradually and engagingly temporal sequences, algorithms, and computational problems allowing student groups to play, explain, create, review, fix, collaborate, and develop perseverance.

Each starts with a challenge or problem and is divided into several stages, allowing different aspects to be addressed through competencies related to computational thinking including problem-solving, idea communication, and expression, while integrating other curricular disciplines.

The activity sequences that comprise the stages include practical suggestions for teachers on group organization, using various resources, and interdisciplinary work.

Group management

The students move, explore, experiment with possibilities, and create collaboratively throughout the sequences. The work teams and the implementation will be determined by the characteristics of each group (number of students, available spaces, devices, etc.) and the pedagogical intentions of each teacher.

Materials

Each sequence incorporates everyday classroom materials and tablets with the ScratchJr application. At the end of the document, teachers will find images for different moments in the activities, mainly from the ScratchJr blocks, which can be printed in the desired size.

Assessment

Assessment is a critical component of the teaching and learning process. It provides opportunities to review ideas, content, and skills, and adjust and modify educational practices, directly impacting the students' learning outcomes.

At the end of the activities, resources are made available to each teacher for conducting self-assessment and evaluation activities, as a guide or starting point for observation and reflection.

Screens and Childhood

Young children's time front of screens (computers, tablets, etc.) has been a concern in the health field. To avoid negative effects, this document follows the guidelines developed by the American Academy of Pediatrics (2016), which emphasize minimizing screen use in early childhood, prioritizing educational content when using technological tools, and ensuring adult or educator supervision (Estefanell, 2021).

Supercircuit

Implementing a circuit with animations enables play-based exploration of an everyday activity.

Stage 1

- **Say It with a Song**
  Represent sequences of sounds (onomatopoeias) and movements presented in a song.

Patterns - Modularity - Sequences – Block-based Programming

Stage 2

- **Shall We Draw?**
  Identify and draw the sequence of actions that make up an activity using ScratchJr.
- **Ideas in Action!**
  Create a simple program in ScratchJr to animate a sequence of images.

Patterns - Modularity - Sequences – Block-based Programming

Stage 3

- **Shall We Play?**
  Create and play with a circuit (game path) that contains all the developed animations.

Patterns - Modularity - Sequences – Block-based Programming

Tell Me a Story

Design and Development of an Interactive Animation

Stage 1

- **Who is the Character?**
  Represent the main characteristics of a story´s character through drawing.

Abstraction - Modularity - Representation – Block-based Programming

Stage 2

- **My Character Moves**
  Represent interactions and movements by a character from the story through an animation in ScratchJr.

Abstraction - Modularity - Representation – Block-based Programming

Stage 3

- **Animated Story**
  Incorporate interactive aspects into the animation.

Abstraction - Modularity - Representation – Block-based Programming

Step by Step

Design and development of an animation that adds movement to different images representing a landscape..

Stage 1

- **Discover It!**
  Create a sequence of instructions using arrows that indicate movements (move forward and turn) to reveal the representative image of a specific landscape.

  Sequence - Algorithm - Representation - Block Programming

Stage 2

- **What is the Path?**
  Create a sequence of instructions using arrows to navigate from a starting point to a destination within a grid.

  Sequence - Algorithm - Representation - Block Programming

Stage 3

- **Animated Landscape**
  Represent and animate a landscape by giving movement to its elements through block-based programming.

  Sequence - Algorithm - Representation - Block Programming

Up and Down

Creation of a simple program to predict outcomes in a game.

Stage 1

- **Jump!**
  Discover how many times a ball bounces up and down while a song plays.

Sequence - Algorithm - Representation - Block Programming

Stage 2

- **How do I Find It Out?**
  Predict the number of jumps that can be made during a song through an animation in ScratchJr.

Sequence - Algorithm - Representation - Block Programming

Stage 3

- **Everything we Learned**
  Create an animation in ScratchJr to predict the result of a race.

Sequence - Algorithm - Representation - Block Programming

Supercircuit

Creation of a circuit of programmed animations to discover a represented daily activity.

In this activity, students create a circuit of animations representing everyday activities. The designed animations are integrated into a game circuit developed collaboratively, which challenges participants to discover the activity represented in each of them.

The development of the activity allows linking physical activities, communication strategies, anticipation, and the creation of sequences with block programming using ScratchJr.

Summary

Supercircuit

Implementation of a circuit with animations to discover an everyday activity.

Stage 1

- **Say It with a Song**
  Represent sequences of sounds (onomatopoeia) and movements in a song.

Patterns - Modularity - Sequences - Block Programming

Stage 2

- **Shall we Draw?**
  Identify and draw the sequence of actions that make up an activity using ScratchJr.
- **Ideas in Action!**
  Create a simple program in ScratchJr to animate a sequence of images

Patterns - Modularity - Sequences - Block Programming

Stage 3

- **Shall we Play?**
  Create and play with a circuit (game path) that contains all the developed animations.

Patterns - Modularity - Sequences - Block Programming

Competencies and Contents of the Integrated Basic Education Program in Computer Science and Educational Technology by section.

STAGE 1: Say it with a song

Section 1

Specific competencies

- CE5.3

Contents

- Computational thinking.
- Ordered sequences in play-based environments.
- Cause-and-effect logic in actions.

Section 2

Specific competencies

- CE5.3

Contenidos

- Computational thinking.
- Computational thinking strategies (with and without digital technologies).
- Ordered sequences in play-based environments.

Music · Physical Education · Spanish Language

STAGE 2: Shall we draw?

Section 1

Specific competencies

- CE1.2; CE2.1

Contents

- Digital literacy.
- Digital image: recording or production.
- Computational thinking.
- Computational thinking strategies.

Section 2

Specific competencies

- CE1.1; CE9.1; CE9.2; CE9.3

Contents

- Digital literacy.
- Digital image. Collaborative digital productions.
- Environments and organization of digital information.
- Computational thinking.
- Computational thinking strategies (with and without digital technologies).

Lengua Española · Artes Visuales y Plásticas

TAGE 2: Ideas in Action

Section1

Specific competencies

- CE1.3; CE5.3; CE7.1

Contents

- Computational thinking.
- Ordered sequences in play-based environments.
- Cause-and-effect logic in actions.

Section 2

Specific competencies

- CE5.2; CE5.3; CE5.5; CE5.6

Contents

- Computational thinking.
- Block programming: simple commands and command sequences.
- Ordered sequences of instructions (algorithms) for problem solving.

STAGE 3: Shall we Play?

Section 1

Specific competencies

- CE1.3; CE5.2; CE9.1

Contents

- Digital literacy.
- Digital devices: recording, information production, exploration, and use (writing, drawing, using applications).
- Collaborative digital productions.
- Computational thinking.
- Ordered sequences in play-based environments.

Section 2

Specific competencies

- CE6.1; CE6.2; CE6.3; CE8.1

Contents

- Digital literacy.
- Digital environments and the organization, storage, and retrieval of information.
- Characteristics and opportunities of using digital resources for communication.
- computational thinking.
- Strategies for computational thinking (with and without digital technologies).

Spanish Language · Visual and Plastic Arts

Say It with a Song

We represent sounds and movements in an organized way

Through a play-based physical activity, each student is encouraged to interpret and represent sequences of sounds (onomatopoeia) and movements from a song.

1 - Looking for a song

Each teacher selects a song that contains a sequence of sounds (onomatopoeia) and movements.

The group of students listens, sings, dances, and enjoys the song repeatedly to learn the lyrics and movements, paying attention to the sequence.

2- Relating and representing

The group is guided through questions to highlight key elements (e.g., professions, fruits, animals, etc.), sounds, and movements mentioned in the song, focusing on their order and repetition.

Students are divided into numbered teams, each assigned an element, sound, and movement based on the order and numbering of the groups. For example, with the song "*El mamboretá,*" each team physically recreates an animal mentioned in the song and presents it at the designated moment.

Next, students are encouraged to modify the sequence by changing the order of the elements, sounds, and movements. For example, with "*El mamboretá*," the anteater could appear first instead of later in the song.

Suggestions

- Coordinate the activity with teachers from other subjects (e.g., Music, Dance, or Physical Education).
- Explore and experiment with other songs and activities that offer new approaches to the concept of sequence.juego.
- Use images or real objects to represent the elements mentioned in the song.
- Organize groups and assign sounds and movements according to the students' preferences and interests.
- Invite families to participate in this experience.

3- Reflection

To wrap up the activity, we review the significance of the order of elements, sounds, and movements in shaping the meaning of the song emphasized.

Guiding questions:

What happens when the sequence is changed? How does the meaning of the song change? (For example, in "El Mamboretá," what happens to the ants when the anteater appears? Does it make sense for the other animals to come help the ñurumí if the anteater appears first?)

Notes

Reviewing the next stage and considering the connection between the song and everyday activities is recommended to maintain a cohesive theme throughout the project.

A smaller number of teams implies less complexity in the activity. It is suggested that the entire sequence be performed with the whole group of students before assigning a sound or movement to each team.

Shall we Draw?

Let's draw everyday activities

As the first step in creating an animation, students identify and draw a sequence of actions from everyday life or a classroom project using ScratchJr.

Selection and communication

The students are invited to create an animation involving movement to a series of images depicting an everyday activity. They create teams and select an animal to represent their identity and name (e.g., frogs). They can choose a different category, keeping in mind that the images must be available in ScratchJr object gallery.

Then, each team selects an everyday activity (e.g., preparing a snack, brushing teeth, getting dressed) and chooses three actions that make it up.

Example: If the chosen activity is brushing teeth, the actions would be:

- Action 1: Open the toothpaste tube.
- Action 2: Place the toothpaste on the toothbrush.
- Action 3: Brush the teeth.

Drawing

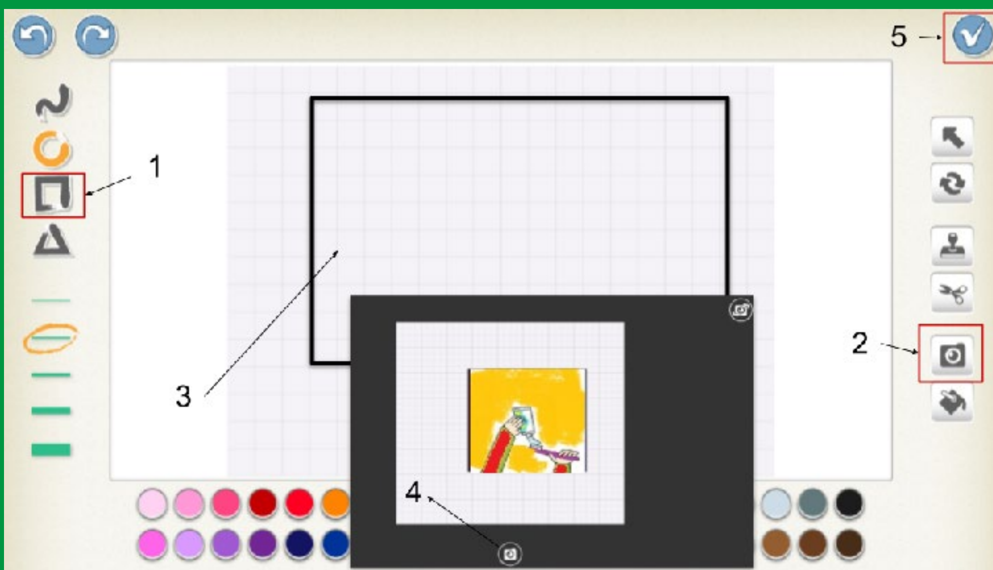
Teams represent the actions through a drawing of each, either digitally using ScratchJr, or on paper. Teachers encourage exploration and creativity using design tools to create the drawings.

Each represented action is uploaded as a page in the app. The photo option in the editor can be used, creating an ordered sequence of images. If students choose to photograph hand-drawn images on paper, they should follow these steps:

Suggestions

· Watch and analyze the video Computational Thinking #PCtubers about the sequence for brushing teeth.

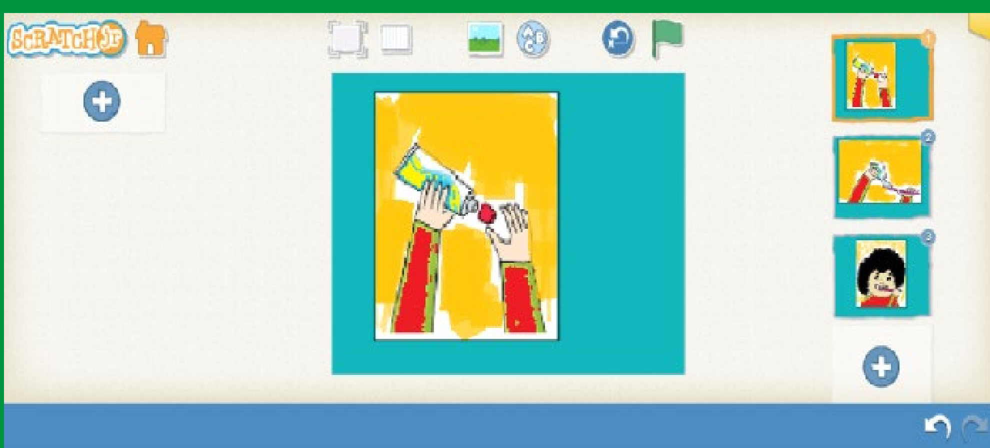
· Record the actions through drawings, collages, clay modeling, written or oral statements, according to your pedagogical interests and the group's potential.



F1. Photo in the image editor.

Action representations are displayed on the right across several pages, arranged in the order in which they were created.

The following image shows an example of tooth brushing represented on paper and added to ScratchJr, with each page photographed and uploaded into the app.



F 2. Pages with photos of created drawings.

The order of the pages is swapped (by dragging them), exploring the possibilities they offer, and emphasizing the order of actions and the changes that occur in the everyday activity when the sequence is modified.

What happens if we change the order of the actions?

Notes

It is important to keep in mind that the animal chosen will later be used in the programming. Therefore, students should select an animal from those available in the gallery: cat, dog, rabbit, hen, pig, horse, zebra, monkey, elephant, giraffe, camel, butterfly, fly, bat, bird, viper, lizard, frog, duck, crab, seahorse, starfish, fish, whale, penguin, and bear.

It is also necessary to consider the number of members and characteristics of the group, as well as available devices and materials before starting, since drawings can be made on paper, collages, or other materials.

To add more complexity to the representation, it is possible to incorporate audio and text into the digital drawings.

Ideas in Action

Animating a sequence of images

Students create a simple program in ScratchJr. to bring movement to a sequence of images.

Recovery and creation

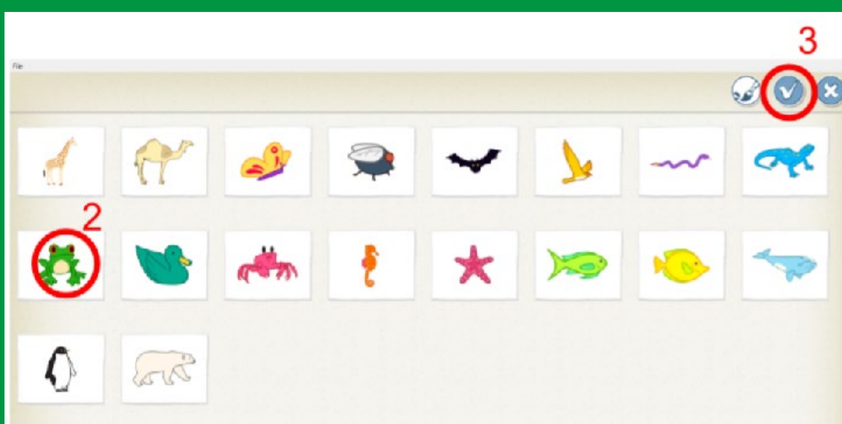
Next, they return to the ScratchJr project using the pages they created to build an animation.

To give identity to the animation, each team incorporates the animal they chose to represent them. To do this, they start on the first page and select the animal from the ScratchJr object gallery. For example, the “Frogs” team would add a frog. They repeat this process on each page of the project, ensuring they use the same object.

Suggestions

- Print and play using the images of the programming blocks in the appendix.

F 3. The symbol indicates the creation of objects.



F 4. Gallery images.

Guiding questions:

What happened when you added a character?

What do you see on the screen?

Do new elements appear? What are they? What will they be used for?

Where else does the character or object you added appear?

Example

F 5. Landscapes and objects in ScratchJr.

Explore Programming Blocks

Before starting the program, students discuss the characteristics of an animation.

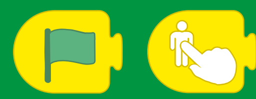
Guiding questions:

What is required for it to be an animation?

What needs to happen with the images?

Each team is invited to explore the programming blocks for the objects. Exploration is guided through questions such as,

F 6. Programming area in ScratchJr



Each team is invited to try these two starting blocks, and then the task of programming the animation is repeated.

It is important to remember that the code is developed in the programming area of each character on each page.

Example: To move from page 1 to page 2, it is suggested to use this code with the character from page 1:

This procedure is then repeated for the character on page 2 using the corresponding code, and so on if more pages are added.

Each team tests their program.

Guiding questions:

What did you observe?

Does it work as you expected?

It is suggested to explore the following block:

What will its function be?

Where would you place it?

Each team tests the program by including this block.

What changes do you observe?

What is it for?

Teams are invited to change the number in the block to a different value and test the program again.

What happened?

Reflection

Teams discuss the programming they have done and the different blocks they used, analyzing both the challenges and successes they encountered throughout the process.

What did you learn?

What did you enjoy the most?

What difficulties did you face? How did you solve them?

Can you improve the program? For what purpose? How?

Time is provided for debugging, if necessary.

Notes

Reusing code to program the character on each page fosters the development of pattern recognition and generalization.

Shall we Play?

We create and play with an animation circuit.

By applying communication and collaboration skills, students will be able to create and play with a circuit (game path) that contains all the animations developed.

Prepare and build

This is the moment to create the activity circuit. Each station in the circuit will correspond to an animation created by a team. Therefore, if there are six teams, there will be six stations, each with its own device.

In a collaborative process, students and teachers decide on the space to be used, the time allocated to the game, and the different roles to be assigned(e.g., explaining the instructions, keeping the animations in good condition and in their designated place, ensuring that the rules are followed). The information is recorded on a whiteboard or graphic organizer for reference in the next activity.

Play and test

The teams go through the different stations in the established order, attempting to guess the depicted in each animation. After recreating the game, they make any necessary adjustments (game rules, path, program details, etc.).

Share

Other groups from the educational community (another class, family members, neighbors, etc.) are invited to play. To begin the activity, introduce the game objective: identifying the everyday activity represented in each animation.

Suggestions

- Make more changes to the order of the sequence to increase the complexity of the game.

The members of each team stay by their devices with their animation to inform each player whether their answers are correct and if they can proceed to the next station in the circuit.

The activity concludes with feedback from the guests, guided by questions generated by the students. Participants are then invited to share their opinions and experiences (on a poster, post-it notes, bulletin board, whiteboard, or in a forum ian the CREA classroom, etc.).

Closure

At the end of the activity, students review the circuit, read feedback left by the guests, and reflect on the entire process.

As a group, they discuss new alternatives:

What would you change? Why?

What did you learn?

What other uses can this programming have?

In what other situations could I use it?

Notes

Creating more than one circuit with fewer sequences allows multiple participants to play simultaneously.

Appendix

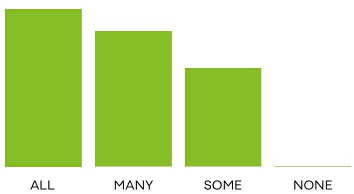
Questions for teacher reflection

Could you replace the analysis of everyday activities with other actions or moments framed within a classroom project?

How could you continue addressing the concepts of sequences, patterns, and modularity?

What positive aspects can you identify when using the digital environment within the activity?

Self-assessment

HOW DID I FEEL WHILE CARRYING OUT THE ACTIVITY? Select the image that represents your emotions.  	HOW MANY ACTIVITIES DID I PARTICIPATE IN?  	DID YOU MANAGE TO CREATE THE ANIMATION YOU WANTED? 1- No 2- Partly 3- Yes
Student 1		
Student 2		

Observation Items for Process Evaluation

Observation	Aspects to consider within each item	Examples of guiding
Pattern Identification	Recognizable, repetitive, and predictable structure; it can be a visual model, a series of events, a mathematical structure, or even a behavior created to solve a situation or problem.	*Is it possible to identify repeating patterns of words, sounds, and/or movements in the songs used that serve a specific purpose, such as representing the characteristics of an animal?*
Sequence	Understanding the importance of the order in which elements are organized within a sequence.	*Do they recognize the importance of the order of actions? (For example, in 'Shall we play?')*
Design of a project	Identify a goal. Abstraction of the main characteristics of a problem and its representation. Implementation of technological solutions.	*Do they recognize what each participant needs to achieve? (For example, in "Shall we play?")* *Did they represent the relevant actions that make up the chosen activity? (For example, in "Shall we draw?")* *Did they overcome obstacles and use strategies to automate the animation of the activity? What were they? (For example, in "Ideas in action!")*
Block programming	Identification of different programming blocks and their function. Creation of a simple program to achieve the objective.	*Did they manage to identify the function of the blocks? (For example, in "Ideas in action")* *Did they manage to reach the planned programming? (For example, in "Ideas in action")*

Tell me a story

Design and development of an interactive animation.

An engaging and fun journey challenges both teachers and students to create an animation that represents characters and special moments from an adventure story. This activity aims to approach algorithms through a block programming sequence and establish cause-and-effect relationships between actions.

Through a sequence of activities organized in three stages, students select and organize relevant information, explore stories with different formats and characteristics, and design and implement a programming code that generates interaction between the character and the story created by their team.

Summary of the Activity

Stage 1

- **Who is the Character?**
  Represent the main characteristics of a character from a story through a drawing.

Abstraction - Modularity - Representation - Block programming

Stage 2

- **My Character Moves!**
  Represent interactions and movements performed by a character from the story, through an animation in ScratchJr.

Abstraction - Modularity - Representation - Block programming

Stage 3

- **Animated Story**
  Integrate new elements and interactive aspects into the created animation.

Abstraction - Modularity - Representation - Block programming

Competencies and Content of the Computer Science and Educational Technology Program

STAGE 1: Who is the character?

Section 1

Specific competencies

- CE1.1

Contents

- Digital Literacy.
- Digital Image: Recording or Production.
- Computational thinking.
- Computational thinking Strategies (with and without digital technologies).

Section 2

Specific competencies

- CE1.1; CE9.1; CE9.2; CE9.3

Contents

- Digital Literacy.
- Digital Image: Recording or Production.
- Collaborative Digital Productions.
- Digital environments and the organization, storage, and retrieval of information.
- Computational thinking.
- Computational thinking Strategies (with and without digital technologies).

Spanish Language: Speaking, Reading - Visual and Plastic Arts

STAGE 2: My Character Moves!

Section 1

Specific competencies

- CE1.3; CE5.3; CE7.1

Contents

- Digital Literacy.
- Digital Image: Recording or Production.
- Iconographies: Relationship Between Icons andTheir Functions in Digital Tools.
- Digital Sound: Recording or Production.
- Collaborative Digital Productions.
- Computational thinking.
- Ordered sequences in play-based environments.
- Cause-and-effect logic in actions.

Section 2

Specific competencies

- CE5.3; CE5.5; CE5.6

Contents

- Digital Literacy.
- Characteristics of Digital Content in Different Formats (games, videos, among others).
- Computational thinking.
- Block Programming: Simple Commands and Command Sequences.
- Ordered Sequences of Instructions (Algorithms) for Problem Solving.

Spanish Language: Speaking, Reading.

STAGE 3: Animated Story

Section 1

Specific competencies

- CE1.3; CE5.3; CE7.1

Contents

- Digital Literacy.
- Digital Devices: Digital Image: Recording or Production.
- Iconographies: Relationship Between Icons and Their Functions in Digital Tools.
- Digital Sound: Recording or Production.
- Collaborative Digital Productions.
- Computational thinking.
- Ordered sequences in play-based environments.
- Cause-and-effect logic in actions

Section 2

Specific competencies

- CE1.1; CE5.2; CE5.3; CE5.5; CE5.6

Contents

- Digital Literacy.
- Uses of Different Languages in Digital Environments.

  Characteristics of Digital Content in Different Formats (games, videos, among others).
- Computational thinking.
- Block Programming: Simple Commands and Command Sequences.
- Ordered Sequences of Instructions.

Spanish Language: Writing

Who is the Character?

Selecting and representing a character from a story.

Reading stories and digital tools complement and enhance each other to represent characters through digital drawing. Based on a story or tale, students are invited to select and represent a character in Scratch Jr.

Listen and Select

Each teacher and their group, selects an adventure story, either in physical or digital format, to read.

Afterward, they discuss which character or characters they found most interesting. Organized into small teams, students select a character to be represented and animated.

What characteristics or actions performed by the selected character did you like the most?

Is the character the main one? Why?

Identify and Represent

The teams digitally draw the character as an object in ScratchJr (Fig. A). They can start with a drawing on paper (which can later be photographed with ScratchJr, Fig. B), or use the drawing tools in the available

(Fig. A)

(Fig. B)

(Fig. C)

Suggestions

- Select stories that your students can understand or will read in upcoming activities..
- Find a diverse collection of uppercase print stories in Biblioteca País, to access multiple copies of the same book allowing the entire group to read simultaneously in consecutive stages..
- Encourage students to make predictions and inferences, then revisit the book to verify their ideas.
- Provide younger students with materials such as crayons and modeling clay to represent the character.
- Explore the guides and suggestions available in the app to learn more about the editing and drawing tools.
- Organize a character presentation with the group.
- If the group wishes, invite families to participate.

Tools for creating objects and pages

Once the character has been drawn, the teams create the setting where their favorite moment of the story takes place. To do this, they use editing tools, take a photo of their drawing on paper, or select a pre-designed background from the ScratchJr. gallery. The background or setting should be added as a page in the animation.

F 1. Objects and pages

Communication

Each group presents their work, discussing the tools used, the challenges faced, and the discoveries made throughout the creative process.

What character traits did they consider when drawing it? Did the drawing tools allow them to create the character as they envisioned? Why?

My Character Moves!

Animating the character

Students create an animation in ScratchJr to represent the interactions and movements of a character from the story read.

Think

The group revisits the previous activity, recalling the character and the chosen moment from the story. They act it out through role-playing.

Next, they reopen the project in ScratchJr, and teams discuss possible strategies to bring movement to the created image. Using the existing character as a starting point, they imagine and create interactions with new characters or the environment.

Who could our character interact with?

Which blocks allow us to create an interaction?

How can we add voice to the characters?

How do we create a new scene?

How can the character move to another scene?

Animate

Teams experiment with different animations in ScratchJr, exploring the available blocks to bring their characters to life. Teachers guide each team, reminding them of the activity's goal: to animate the character or create interactions between characters, e.g., a dialogue.

Suggestions

- Select stories that your students can understand or will read in upcoming activities.
- Encourage students to make predictions and inferences, then revisit the book to verify their ideas. With younger students, provide materials such as crayons and modeling clay to represent the character.
- Explore the guides and suggestions available in the app to learn more about the editing and drawing tools.
- Organize a character presentation with the group.
- If the group wishes, invite families to participate.

Why do some blocks have different shapes than others?

Explore two different movement blocks.

What do the symbols on these blocks mean? How are they different?

Example

The image shows a possible programming sequence that allows the dog to move toward the frog and say “hello.”

F 2. Dialogue and movement blocks

In this second example, the code shows how the dog moves toward the frog, says “hello,” and then continues walking until the scene changes.

F 3. Movement, dialogue, and page change blocks

Reflection

A brief presentation session is held where each team showcases and explains how they tackled the given challenge, highlighting the sequence of actions taken to achieve the final result.

At the end of the presentation, the group discusses ScratchJr's possibilities for animating the story. They reflect on the elements used to represent character dialogue, the programming blocks enabling these interactions, and how they modified or improved their programs.

Notes

Character dialogue can be recorded by students as audio, without requiring written text.

To increase the difficulty level, students can create a dialogue where each character has more than one written statement.

To simplify the activity, the challenge can be limited to interacting with another character or object without adding a new page.

Suggestions

• Emphasize the importance of including their names as authors of the programs they create, as well as the name of the author of the story they read.

• Before exploring objects and blocks that enable interaction between the user and the program, introduce the concept of cause and effect to help students better understand interaction.

• For example, use outdoor games like tag (in its different variations), races, or jumps based on given instructions. It is important for students to understand that an action (such as a command, touch, or sound) must occur for another action to take place.

• Watch the following video to learn more about the tools available for adding text to animations.

• If a large screen is available in the classroom, use it to display the application, demonstrate the available tools, and collaboratively test different codes to achieve the objective.

Animated Story

Let's create an interactive animation

New interactive elements are added to the animation through different and progressive approaches to ScratchJr digital environment.

Retrieval

The previous activity, the achieved objective, and the lessons learned are revisited by recovering the saved animation.

The group then reflects on possible improvements and interactive elements that could be incorporated into the animation created in the previous activity, applying everything learned so far.

Design of Interactions

The group reorganizes into teams to enhance their animations. For example:

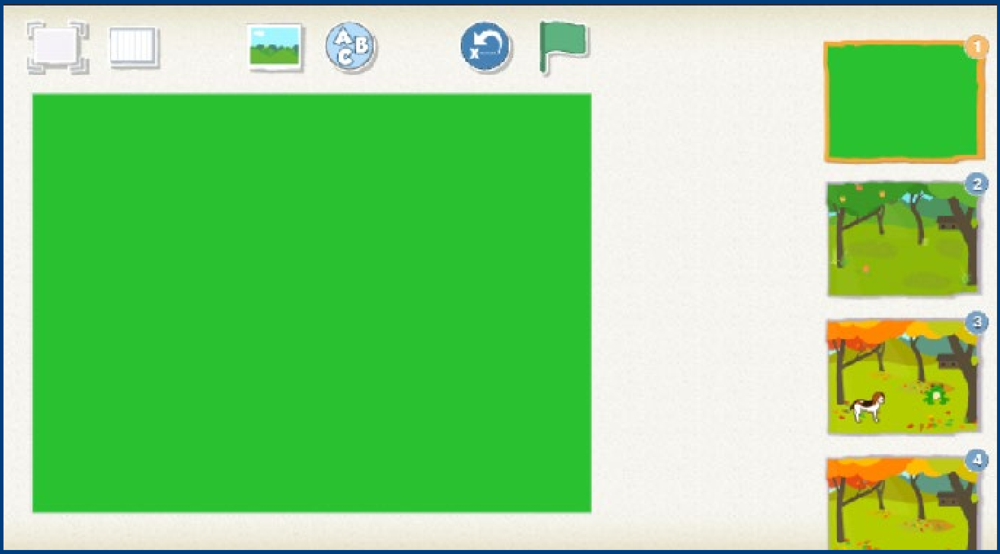
How do we know which story and characters the animation refers to?

How can we include this information?

Once the possible strategies and decisions have been discussed, teams select programming blocks and experiment with algorithms, applying knowledge such as adding a new page, inserting a title, and including dialogue for their characters.

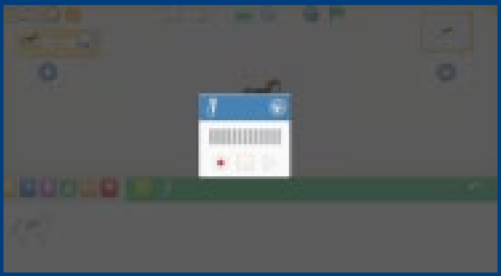
This is illustrated as follows:

- Add a new page, customize it with the desired colors and shapes, and place it (drag it) to the first position.



F 4. New page placed in the first position.

- Add the story title in different formats.
- Record audio with the story title.



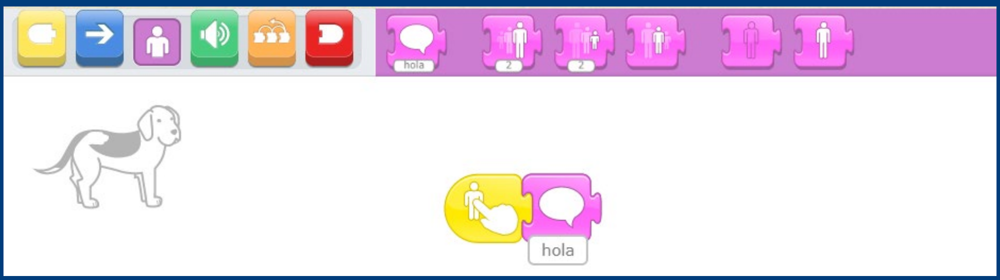
F 5. Tool for recording audio in ScratchJr

- Create code that allows playback when clicking on a button.

F 6. Event block and sound block.

This programming can also be used to add more information about the story.

- Display the characters' names when clicked on.



F 7. Event block and looks block.

- Next page

F 8. Event, sound, and end blocks.

Code that allows you to hear a message indicating that by pressing the button, you can access the animation and move to the next page.

Present

A presentation session is organized where each team presents and explains how they tackled the given challenge.

Appendix

Questions for Reflection

What changes would you suggest to achieve the set objective, but using a different type of discourse or genre?

What communication skills help you approach these activities?

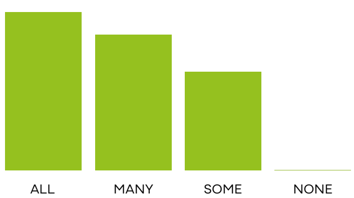
What types of learning are supported by incorporating audio and images in motion?

What skills will your students develop by creating code that allows a character to move?

Self-assessment

¿HOW DID I FEEL WHILE CARRYING OUT THE ACTIVITY?
Select the image that represents your emotions.

¿HOW MANY ACTIVTIES DID I



¿DID YOU MANAGE TO CREATE THE ANIMATION YOU WANTED?

1- No
2- Partly
3- Yes

Student 1		
Student 2		

Observation Items for Process Evaluation.

Observation items	**Aspects to consider within each item**	**Examples of guiding questions**
Sequence	Sequencing of blocks that achieve the desired objective.	*Do the selected and sequenced blocks allow the desired action to be executed?*
Abstraction	Selection and representation of information to achieve a set objective.	*Is at least one of the characteristics of the selected character distinguishable in the drawing?*
Representation and block programming.	Interpretation and use of symbols in block programming language. Appropriate selection of blocks.	*Is it possible to observe an appropriation of the tools and code for communicating ideas?* *Is there a correct use of the programming blocks?*
Interactive story	Creation of an animation related to the story.	*Is the animation consistent with the story created?*

Step by Step

Design and development of an animation that allows movement of different images representing a landscape.

Movements in space and on a plane are combined with knowledge of landscapes and locations to create an engaging animation by designing an animated landscape in ScratchJr.

The activity allows students to interpret and create sequences of instructions to develop an algorithm, establishing causal relationships between actions, and incorporating movement into an image through block programming.

Summary of the Activity

Stage 1

- **Discover It!**
  Create a sequence of instructions using arrows that indicate movements (move forward and turn) to discover the representative image of a specific landscape.

Sequences - Algorithm - Data analysis and organization - Block programming.

Stage 2

- **What is the Path?**
  Create a sequence of instructions using arrows to navigate from a starting point to a destination within a grid.

Sequences - Algorithm - Data analysis and organization - Block programming.

Stage 3

- **Animated Landscape**
  Represent and animate a landscape by giving movement to its elements through block-based programming.

Sequences - Algorithm - Data analysis and organization - Block programming.

Competencies and Content of the Computer Science and Educational Technology Program

STAGE 1: Discover It!

Section 1

Specific competencies

- CE1.3; CE5.3

Contents

- Computational thinking.
- Ordered sequences in play-based environments.

Section 2

Specific competencies

- CE5.2; CE5.5; CE5.6

Contents

- Computational thinking.
- Ordered sequences in play-based settings.

Physical Education · Spanish Language · Geography

ETAPA 2: ¿Cuál es el camino?

Section 1

Specific competencies

- CE1.3; CE5.3

Contents

- Computational thinking.
- Ordered sequences in play-based environments.

Section 2

Specific competencies

- CE5.2; CE5.5; CE5.6

Contents

- Computational thinking.
- Ordered sequences in play-based contexts.

Physical Education · Spanish Language · Geography

STAGE 3: Animated Landscape

Section 1

Animated Landscape

- CE1.3; CE5.3

Contents

- Digital literacy.
- Digital image: recording or production.
- Iconographies: relationship between icons and their functions in digital tools.
- Collaborative digital productions.
- Computational thinking and robotics.
- Ordered sequences in play-based environments.
- Logic of cause and effect of actions.

Section 2

Specific competencies

- CE5.3; CE5.5; CE5.6

Contents

- Digital literacy.
- Uses of different languages (oral, iconic, gestural, written) in digital environments. characteristics of digital content in different formats (games, videos, and others).
- Computational thinking.
- Block programming: simple commands and command sequences.
- Ordered sequences of instructions (algorithms) for problem solving.

Geography

Discover It!

We discover landscapes by following instructions.

Students interpret and execute instructions to achieve a goal, using different symbolic representations and spatial concepts.

Interpretation

The group observes and analyzes images of different places or geographical landscapes (urban or rural).

Examples:





Each teacher selects one of the images and invites students to discover which one was chosen by following a sequence of provided clues.

The clues refer to the main characteristics of the selected landscape and are presented gradually.

Example:

If the following image was selected:

Suggestions

- Decide in advance which places or landscapes you want to observe and what aspects and components you will emphasize according to your class intentions and projects.
- Take walks around the neighborhood or city, invite families, and take representative photographs of the observed places (squares, parks, the school, neighborhood block, etc.)
- When providing clues, you can incorporate spatial concepts and cardinal points according to your pedagogical goals and the group's potential.
- Explore the Desde mi cuarto application with your students; it is available on first-cycle tablets. The challenges in the Aula game allow for continued practice with sequenced instructions and clues.

The teacher provides clues, such as:

"The selected place is very crowded."

"Most of its visitors are young people and children."

"This place can be used for playing and sports."

The whole group participates by analyzing the clues and trying to determine which image they refer to. Additional clues can be added as needed.

Understanding instructions

In a new activity, three of the images are selected and placed in different locations in the classroom or playground. The images should be printed in a size that ensures clear visibility for the entire group.

Once again, the teacher selects one of the printed images and invites students to discover their choice. This time, the given instructions describe spatial movements that a small group of students must follow to reach the selected image.

It is important to establish a clear starting point before executing the movements and to give instructions one at a time (one instruction, one movement).

In a new variation of the game, instructions are communicated using symbols as an alternative way to represent movements; in this case, arrows. These are shown and analyzed beforehand.

Example: Each teacher presents the symbol (Fig. A) to the group so they can interpret and execute the movement.

Fig A

Next, two groups of students are selected. The first group chooses the landscape, while the second group follows the instructions using the symbols.

Analysis and evaluation

At the end of the activity, students reflect on their achievements and challenges discussing strategies for improvement in the next game session.

They also analyze the use of graphic symbols (arrows) and spoken language to communicate ideas.

Notes

Proposing new movement and search games both inside and outside the classroom will help to reinforce and build on the work done in this activity. Executing the activity in larger spaces increases the complexity of the task. At the end of the activity guide, you will find printable materials to support its implementation.

Which is the Way?

We create instructions with blocks.

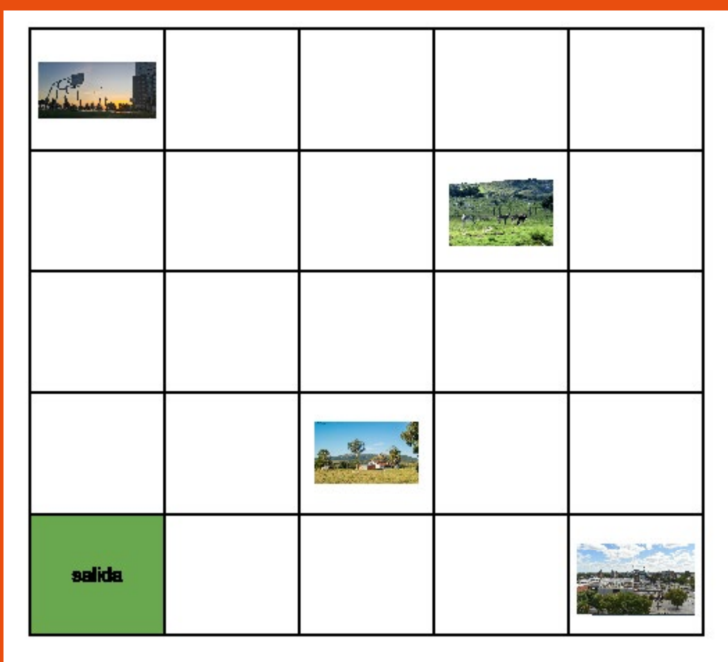
The group of students creates a sequence of instructions using arrows to move within a grid.

Interpreting and understanding

Each teacher presents their students with printed images of the landscapes studied, assigning each one a name (e.g., countryside, house in the countryside, Colonia, city, Montevideo, city center, plaza.) These images are placed within a grid, where each cell is at least 30 cm per side.

The images are distributed across different grid cells, and a set of arrows representing movements is provided. As in the previous activity, a starting point is indicated.

Examples:

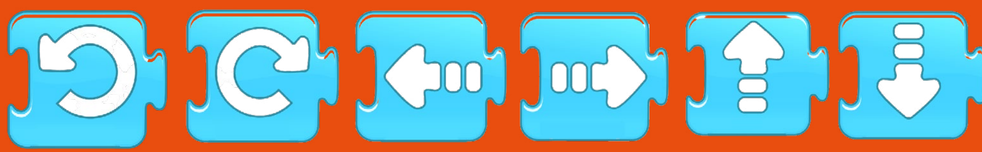


2. Grid with landscape photos

Suggestions

- Revisit the places/landscapes explored in the previous activity.
- Play with the whole group multiple times, creating different paths using the arrows to reach the various images.
- Remember that each arrow represents a movement. You can provide each team with printed arrows, or they can draw them on paper.
- Assign locations to each team using images, written words, or a secret oral message. Using all three methods simultaneously will make the activity more accessible.
- Invite your students to come up with a unique name for each team.
- Manage time so that at least two teams can present and verify their paths.
- Reflect on alternative routes to reach the goal and the number of steps required.

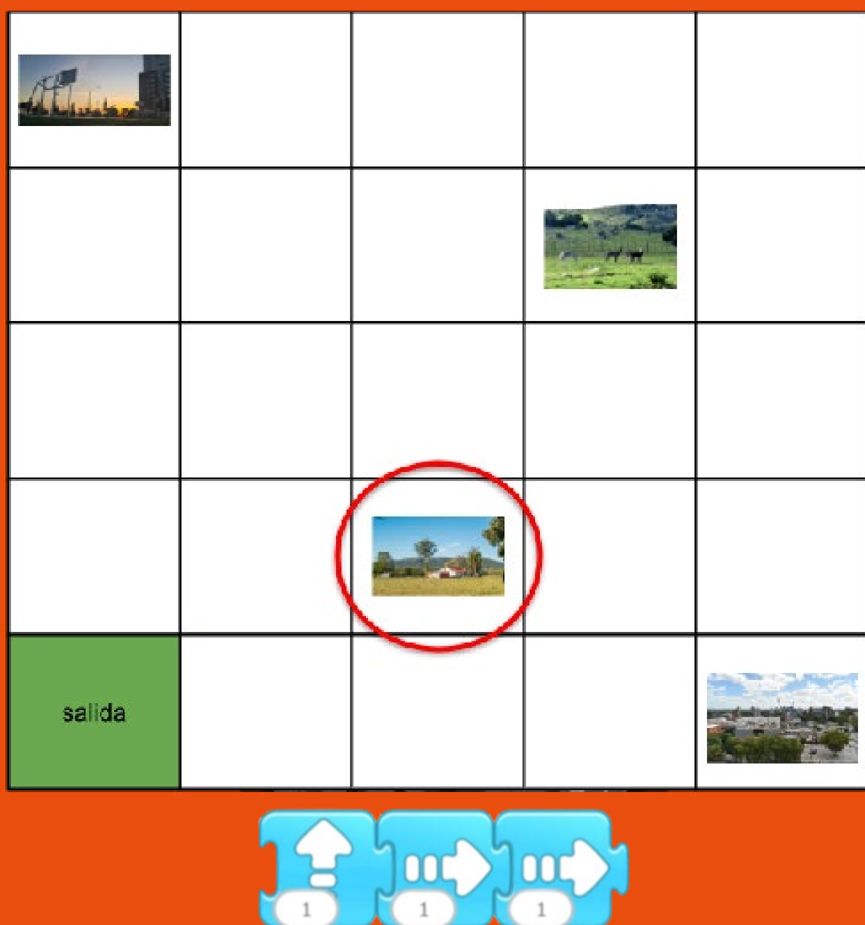
The meaning of each one is analyzed collectively.



F 3. Movement arrows

Next, the group is encouraged to organize into teams for the game *"I'm Going on a Trip to..."*. Each team receives a different destination and the necessary materials to represent the symbols. They then create a sequence of instructions using the symbols to move from the starting point on the grid to the assigned image.

Example:



F 4. Grid with a marked photo and a sequence of symbols indicating movements

The results are shared by indicating a landscape and asking one of the teams assigned to that destination to share the sequence of instructions they created.

To verify the algorithms, someone from the group is asked to follow the steps indicated by the team and check if they reach the correct destination.

Exploration

Once the routes are verified, the cards are removed and placed in different locations within the grid to start a new challenge.

This time, each teacher assigns new destinations, ensuring that each team knows only their own objective.

After the allotted time for creating their routes, the game begins with the phrase: "Where did team... go on their trip?"

The sequences of instructions (algorithms) are presented, analyzed by the whole group, and executed by someone who does not know the destination.

Reflection and execution

Once all teams present and verify their algorithms, students analyze the successes and challenges encountered during the game. Emphasis is placed on the importance of the order in the sequence of instructions and spatial relationships.

Notes

The number of squares in the grid determines the number of instructions required, directly affecting the difficulty level proposed for the group of students.

Larger arrows and the possibility of interlocking them help younger students better understand the sequence of instructions.

At the end of the activities, you will find printable materials to support implementation.

Animated Landscape

We give movement to an image using block-based programming.

Students represent and animate a landscape using block-based programming.

Ideas to try out

- Work with landscape elements: artificial and natural, static and moving.
- Refer to previous activities where strategies are gradually and sequentially developed to help students become familiar with the design tools available in the application.
- If appropriate and students show interest, continue working on object animation. Encourage them to select and edit pre-designed images from the application's gallery and explore the available blocks.

Selecting, Organizing, and Representing

Each teacher discusses with the students the characteristics of the landscapes explored.

After the analysis, the group is divided into small teams and tasked with selecting and animating one of the landscapes using ScratchJr tools.

Together, they establish the scope of the animation:

What does it mean for the landscape to be animated?

Will the entire landscape move, or only certain elements?

What will happen when the elements move?

The teams decide which landscape features to represent, which elements to include, and which ones will be animated. The background of the landscape is represented as a page, while the elements are added as objects within the application.

Teams can draw on paper (or use modeling clay or other materials), take a photo with the ScratchJr editor, or create their designs digitally using the drawing tools and image gallery.

F 5. Example of animation with an animated landscape

Creation of a Code

The teams explore the possibilities ScratchJr offers for moving and positioning objects through an ordered sequence of blocks.

Example:

Below are possible codes that enable object movement.

Both codes allow the character to move three steps forward, wait, and return to the starting point.

Reflection

The teams present their animations they created and explain the process of designing and implementing their ideas.

What landscape did you animate?

Why did you select those elements?

What changes occur in the landscape when movement is applied?

During the presentations, the animations are analyzed collectively. The group reflects on which programming blocks in ScratchJr allow different movements and how they need to be sequenced to achieve different objectives.

If we want the character to move, what would be the recommended sequence of blocks?

If we change the direction the object is heading, does the sequence of blocks change or is it the same?

How do we make an object shake in place?

Can we reach the same solution using two different programming approaches?

Notes

The scene and the elements can be selected from the application's image gallery or explicitly drawn. The latter option allows for a deeper exploration of the characteristics of the landscape but requires more time to create.

The number of animated elements will depend on each team's skills, their understanding of spatial orientation, the structured sequencing of instructions, and their ability to use the ScratchJr tools.

Appendix

Questions for Reflection

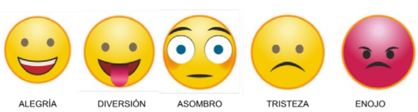
What other themes do you think could be explored using the dynamics developed in this activity?

What modifications would you make to better adapt the activities to the needs and characteristics of your group?

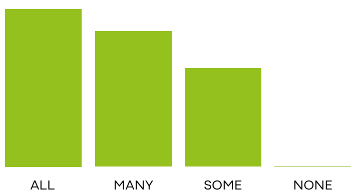
How could you further develop the symbolic representation of ordered instructions?

Self-assessment

HOW DID I FEEL WHILE CARRYING OUT THE ACTIVITY?
Select the image that represents your emotions



HOW MANY ACTIVITIES DID I PARTICIPATE IN?



¿DID YOU MANAGE TO MAKE THE ANIMATION YOU WANTED?

1- No, none
2- Some
3- Many

Student 1		
Student 2		

Observation items for process evaluation

Observation items	Aspects to consider within each item	Examples of guiding questions
Interpretation and creation of sequenced instructions.	Interpretation and execution of a sequence of instructions. Development of ordered instructions to achieve a goal.	*Do they understand the instructions given to reach a physical or digital object?* *Do they create instructions to discover the designated landscape?*
Development of a simple code	Development of a sequence of ordered instructions using graphical signs.	*Do they create a sequence of ordered arrows indicating a possible path to reach the landscape?*
Create a block-based program to give movement to a digital object.	Creation of a sequence of instructions following the rules of block-based programming language.	*Do they build a sequence of instructions using programming blocks in ScratchJr?*

Up and Down

Creation of a simple program that allows predicting quantities.

An animation is developed to predict quantities and solve problems in a playful situation.

The activity involves using block-based programming, verifying the effectiveness of the code, and reflecting on the limitations and functions of technology in today's society.

Summary of the Activity

Stage 1

- **Jump!**
  Discover the number of times a ball bounces up and down while a song plays.

Analyze, interpret, and represent information - Patterns - Algorithm - Block-based programming - Repetition loops

Stage 2

- **How do I Find It Out?**
  Predict the number of jumps that can be made during the duration of a song through an animation in ScratchJr.

Analyze, interpret, and represent information - Patterns - Algorithm - Block-based programming - Repetition loops

Stage 3

- **Everything We Learned!**
  Create an animation in ScratchJr to predict the result of a race.

Analyze, interpret, and represent information - Patterns - Algorithm - Block-based programming - Repetition loops

Competencies and Contents of the Computer Science and Educational Technology Program

ETAPA 1: Jump!

Section 1

Specific competencies

- CE5.2; CE5.3; CE7.1

Contents

- Computational thinking and robotics.
- Ordered sequences in play-based environments.
- Cause-and-effect logic in actions.
- Decomposition of simple problems.
- Digital citizenship.
- The social uses and functionalities of technology: sharing, playing, learning, communicating, and problem-solving.

Section 2

Specific competencies

- CE4.3; CE5.1; CE5.2; CE5.3; CE5.4; CE5.5; CE5.6

Contents

- Computational thinking and robotics.
- Expression of solutions with symbols.
- Block-based programming: simple commands and command sequences.
- Ordered sequences of instructions (algorithms) for problem-solving.
- Strategies for computational thinking.
- Digital citizenship.
- Characteristics of computational language and its relation to other languages.

Mathematics · Physical Education

STAGE 2: How do I Find It Out?

Section 1

Specific competencies

- CE5.2; CE5.3; CE7.1

Contents

- Computational thinking and robotics.
- Ordered sequences in play-based environments.
- Cause-and-effect logic in actions.
- Decomposition of simple problems.
- Digital citizenship.
- Social uses and functionalities of technology: sharing, playing, learning, communicating, and problem-solving.

Section 2

Specific competencies

- CE4.3; CE5.1; CE5.2; CE5.3; CE5.4; CE5.5; CE5.6

Contents

- Computational thinking and robotics.
- Expression of solutions using symbols.
- Block-based programming: simple commands and sequences of commands.
- Ordered sequences of instructions (algorithms) for problem-solving.
- Computational thinking strategies.
- Digital Citizenship.
- Characteristics of computational language and its relationship with other languages.

Mathematics · Physical Education

STAGE 3: Everything We Learned!

Section 1

Specific competencies

- CE5.2; CE5.3; CE7.1

Contents

- Computational thinking and robotics.
- Ordered sequences in play-based environments.
- Logic of cause and effect in actions.
- Decomposition of simple problems.
- Digital Citizenship.
- Social uses and functionalities of technology: sharing, playing, learning, communicating, and problem-solving.

Section 2

Specific competencies

- CE4.3; CE5.1; CE5.2; CE5.3; CE5.4; CE5.5; CE5.6

Contents

- Computational thinking and robotics.
- Expression of solutions using symbols.
- Block-based programming: simple commands and sequences of commands.
- Ordered sequences of instructions (algorithms) for problem-solving.
- Computational thinking strategies.
- Digital Citizenship.
- Characteristics of computational language and its relationship with other languages.

Mathematics

Jump!

We estimate how many times a ball bounces up and down.

Students estimate the number of times a ball bounces up and down on the floor while a song plays.

Representation

Students play in the schoolyard with balls, making them bounce within a time set by their teacher.

They analyze the movements of the balls as they bounce (going up and down) and are encouraged to imitate these movements with their bodies.

What was the starting point?

Where was it going?

How can these movements be represented with the body?

At the end of the game, they discuss and clearly establish the path of the ball as it bounces (going up and down). This movement will be revisited throughout the activity..

Estimation

In a new phase, each teacher presents a song and invites the group to sing and jump, representing the movements described in the song.

The group is asked about the number of jumps possible within the duration of the song, and their estimates are recorded.

How many jumps do you think you made?

Suggestions

- Invite teachers from other subject areas (e.g., Physical Education, Music) to take part in the activity.
- Challenge the group to keep the balls bouncing for a set period of time.
- Carry out as many activities as necessary until students fully understand and can represent the up-and-down movements.
- Share the song with the group and their families so they can continue playing at home.
- Discuss with students the relationship between the two variables (speed and duration of the song).
- Encourage your students to jump to the rhythm of the song.

Verification

The group represents the movements of a bouncing ball throughout the duration of the song, counting the number of jumps they make, landing on both feet.

They compare their estimates by making the differences in the counts visible, highlighting variations in individual results.

Each teacher engages in a discussion with the group about the differences in the counts and how factors like speed and height might influence the total.

Notes

The level of difficulty introduced at this stage, and in the subsequent ones, necessarily requires going through the previous activities or developing the computational thinking skills described at the beginning of the project.

The selection of the song and the number of verses should be based on the group's potential.

Extending the duration of the song will result in variations in the possible number of jumps.

How do I Find It Out?

Anticipating Quantities.

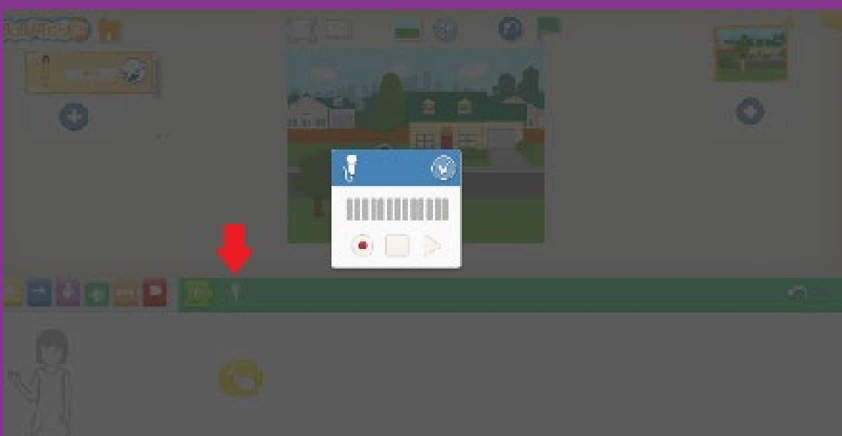
Through block-based programming in ScratchJr, the previous game scenario is represented to predict quantities.

Design

The activity from the previous stage is revisited collaboratively. Srategies used to anticipate the number of jumps are analyzed.

A fragment of the song is selected, and teams record it using ScratchJr.

Example



F 1. Options to record sounds

Each team proposes strategies to predict the number of jumps they can make during the recorded song fragment without actually jumping. How can we discover how many jumps can be made during the song without performing the up-and-down movements?

The students are invited to use ScratchJr as a way to discover it.

Exploring

Each teacher guides the teams to represent a jump through ScratchJr, focusing on the up-and-down movement.

Suggestions

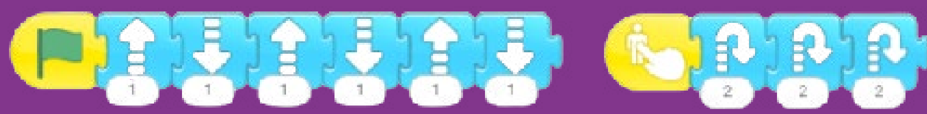
- By this stage, it is essential that students have mastered counting and one-to-one correspondence to cardinalize the number of jumps, establishing the relationship between movement and number.
- Take into account the spatial limitations of the classroom for the collective jumping game.
- Explore ScratchJr's possibilities for recording. Emphasize the importance of saving recordings to replay the song as needed and highlight how technology facilitates this process.
- If resources allow, display different ScratchJr animations on a large screen to model various situations. Review examples for inspiration.
- Guide teams in clearly identifying when the character starts moving—whether by clicking on it or pressing the green flag—and defining its movement (up and down on the screen).
- If possible, share the app screen or print out the blocks to collaboratively create one of the possible codes to solve the challenge.

To do this, they select or draw a character from the gallery and sequence the movement blocks. They apply the knowledge gained in previous activities to collaboratively create a code that makes the character go up and down, simulating a bouncing motion.

Example

When the animation begins (either by clicking the green flag or tapping the object), the previously recorded begins to play, and the object moves up and down repeatedly.

F 2. Possible codes



Verification

Each team shares their complete code, analyzes strategies and areas for improvement, and builds generalizations.

Different ways of representing the jump are shown, analyzing the differences and similarities in the codes.

The teams present the number they calculated as the expected number of jumps and test their prediction by jumping while the song plays.

The group discusses the importance of rhythm, song speed, and jump height and how these variables affect the final result.

Does the number of jumps change if the song plays faster or slower?

If the jumps are higher, will fewer jumps be made?

Why aren’t the quantities found by all the teams the same?

Notes

This stage allows for greater complexity to be added to the previous task. Each group can record the original song so all teams have the same version.

They can sing it during each test or sing it once and record it. This implies that each team will have different timings.

To increase the complexity of the task, students can create a new audio with additional verses recorded, where the previously learned fragment (e.g., the chorus) is repeated at least twice.

Everything We Learned!

We predict results with an animation.

All the knowledge gained in the previous stages and activities is now applied to predict the outcome of a race, using only ScratchJr.

Prediction

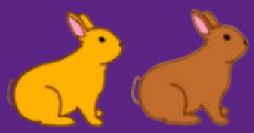
Students are reminded of their previous experience, emphasizing the potential of programming to create models.

Each teacher proposes a new game: predicting the result of a race.

To do this, two characters are presented, both determined to race, where after each step, they take a jump. The following question is posed: If one of the characters always jumps higher than the other, who will reach the finish line first?

The jump race is represented, where different students play and become the characters presented by their teacher. At the end of the representation, the group shares their assumptions and reasoning about the race results, initiating a debate. These contributions help each teacher guide the class toward developing a solution in ScratchJr and testing the ideas proposed.

Example



F 3. Characters of the same size.

Suggestions

- Select animals or characters related to themes or interests of your students.
- Use the printable arrows to collaboratively create the algorithm that will allow each student to perform the required movements in the race before moving on to the application.
- If you want to use the same figure to represent both characters, encourage your students to personalize them with different colors or accessories.

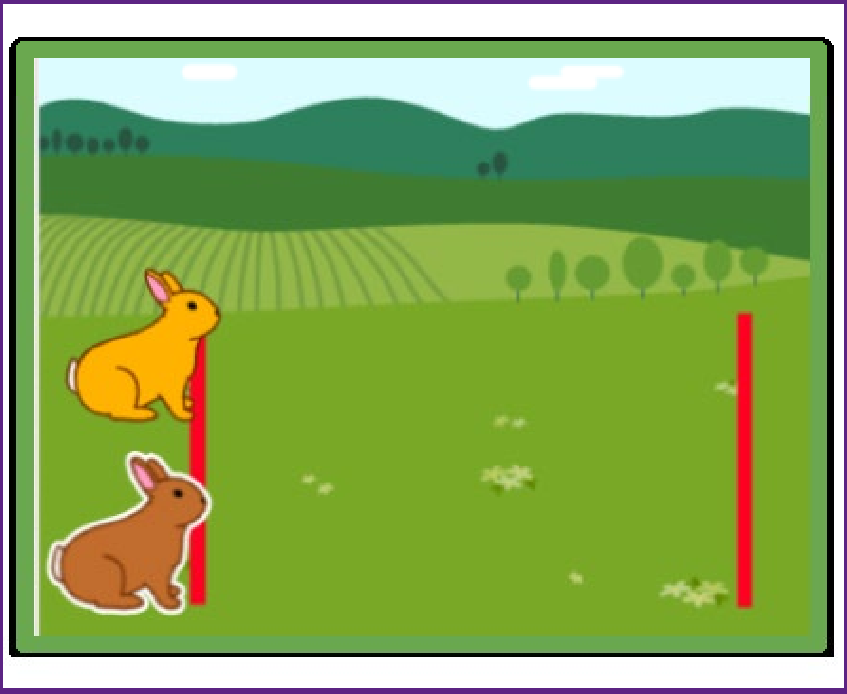
Redefinition

Previous projects are revisited and viewed in the ScratchJr projects section; each developed program's "what" and "why" are analyzed.

F 4. View of all projects created in ScratchJr.

The teams model the situation using ScratchJr programming to find an answer to the given question.

The teams are instructed on the importance of drawing the starting and finishing points to mark the beginning and end of the race.



F 5. Start and finishing points of the race.

Each teacher walks around the teams, asking different questions that help recall strategies and lessons learned in previous activities on programming movements (displacements and jumps).

When will the race start?

Should both characters start at the same time or separately?

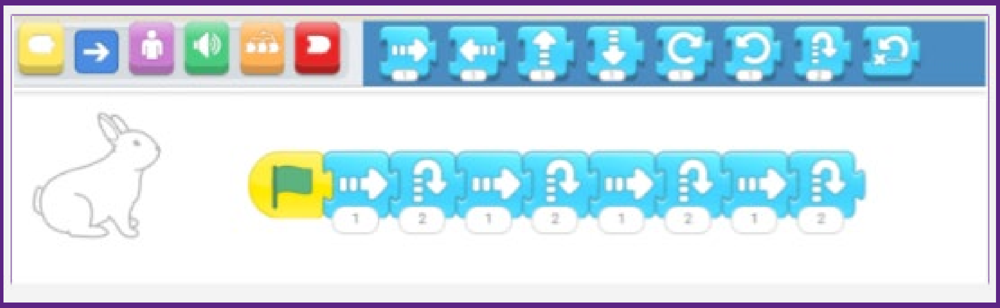
How can an object be made to move one step forward?

Which block will indicate the correct direction?

When should each character stop moving?

Through trial and error, students program the blocks, defining the number of times each movement should be executed.

Example:



F 6. Possible code with repeated movements.

Verification

The teams test the code they created and present their animation, explaining to their classmates the strategies, decisions, and results obtained.

Together with their teacher, they analyze the factors that cause changes in the results:

What happens if the race is longer? What if both jump the same distance?

Does the difference in jumps affect the race outcome? Why?

If characters of different sizes are selected, would the result change?

Based on the created animation, is one character faster than the other?

At the end of the activity, the group engages in a discussion with their teacher about their predictions and possible ways to use technology to model situations.

They reflect on the presence of technology in daily life, factors influencing the results, and the role of people in its implementation. .

Why is it useful to know or estimate a result before testing it?

When and why is it useful to use a computer to estimate or predict a result?

When and where do these predictions and estimations occur? For example, how is it possible to know if it will rain over the weekend?

Reuse

tudents are asked to modify their programs to ensure that one of the characters wins the race.

Which blocks allow modifications?

They are encouraged to make adjustments and explore changes until the objective is achieved.

Notes

Decide in advance whether the characters will be of similar or very different sizes to adjust the difficulty of the task.

Adding more characters or increasing the number of jumps and steps each one takes will also make the task more challenging.

Appendix

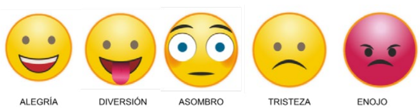
Questions for reflection

What changes would you make to the activities to better adapt them to your students' needs and characteristics while keeping the stated objective and the dimensions of computational thinking in mind?

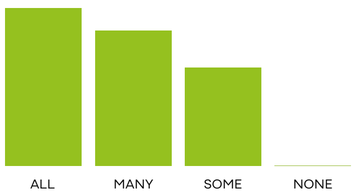
Self-assessment

¿ HOW DID I FEEL WHILE CARRYING OUT THE ACTIVITY?

Select the image that represents your emotions



HOW MANY ACTIVITIES DID I PARTICIPATE IN?



DID YOU MANAGE TO MAKE THE ANIMATION YOU WANTED?

1- No, none
2- Some
3- Many

Student 1		
Student 2		
Student 3		

Observation items for process evaluation

Observation items	Aspects to consider within each item	Examples of guiding questions
Block-based programming	Selection and organization of blocks to achieve a set objective.	*Do the selected blocks represent the given problem situation? For example, "How do I figure it out?" to represent the movement of going up and down.*
Test the code	Verification of the code's effectiveness.	*Does the created code predict the given situation? For example, the number of jumps or which character will reach the finish line first.*

Bonus: Solve different Bebras challenges and sign up with your class for the big Bebras challenge, which takes place worldwide in November.

Learn more about the Bebras challenge here.

References

Administración Nacional de Educación Pública, Marco Curricular Nacional. Transformación Educativa [online]. Uruguay, 2022a. Available online: https://www.anep.edu.uy/sites/default/files/images/Archivos/publicaciones/Marco-Curricular-Nacional-2022/MCN%202%20Agosto%202022%20v13.pdf [Date of last access: 11 de abril de 2023].

Administración Nacional de Educación Pública (2022b). Progresiones de Aprendizaje. Transformación Curricular Integral [online]. Uruguay, 2022b. Available online: https://www.anep.edu.uy/sites/default/files/images/2022/noticias/agosto/220829/Progresiones%20de%20Aprendizaje%202022%20v4.pdf [Date of last access: 11 de abril de 2016].

Administración Nacional de Educación Pública, Programa de Educación Básica Integrada. Ciencias de la Computación y Tecnología Educativa. Tramo 1 [online]. Uruguay, 2023a. Available online: https://www.anep.edu.uy/sites/default/files/images/te-programas/2023/finales/espacios/espacio-tecnico-tecnologico/Ciencias%20de%20la%20Computaci%C3%B3n%20y%20Tecnolog%C3%ADa%20Educativa%20-%20Tramo%201.pdf [Date of last access: 11 de abril de 2023].

Administración Nacional de Educación Pública, Programa de Educación Básica Integrada. Ciencias de la Computación y Tecnología Educativa. Tramo 2 [online]. Uruguay, 2023b. Available online: https://www.anep.edu.uy/sites/default/files/images/te-programas/2023/finales/espacios/espacio-tecnico-tecnologico/Ciencias%20de%20la%20Computaci%C3%B3n%20y%20Tecnolog%C3%ADa%20Educativa%20-%20Tramo%201.pdf [Date of last access: 11 de abril de 2023].

BERS, Marina Umaschi, Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom. Nueva York: Routledge, 2020.

Ceibal, Pensamiento computacional. Propuesta para el aula [online]. Uruguay, 2022. Available online: https://bibliotecapais.ceibal.edu.uy/info/pensamiento-computacional-propuesta-para-el-aula-00018977?locale=es [Date of last access: 11 de abril de 2023].

ESTEFANELL, Lorena, Pantallas en casa: Orientaciones para acompañar una navegación segura en internet. Guía para las familias [online]. Uruguay: Plan Ceibal - ANEP - Unicef, 2021. Available online: https://www.unicef.org/uruguay/media/5011/file/Pantallas%20en%20casa.pdf [Date of last access: 11 de abril de 2023].

RESNICK, Mitchel, "All I really need to know (about creative thinking) I learned (by studying how children learn) in kindergarten", en Proceedings of the 6th ACM SIGCHI Conference on Creativity and Cognition. Washington DC, 2007, p. 1-6.

Unesco, Glosario de términos [online], Media Information Literacy for Teachers, ONAOC, 2011. Available online: http://unesco.mil-for-teachers.unaoc.org/glosario-de-terminos/?lang=es [Date of last access: 11 de abril de 2023].

VILLARROEL, Paola, La construcción del conocimiento en la primera infancia [online]. Quito: Universidad Politécnica Salesiana del Ecuador, 2012. Available online: https://dialnet.unirioja.es/servlet/articulo?codigo=5973111 [Date of last access: 11 de abril de 2023].

Glossary

Abstraction	An intellectual ability that involves isolating an element from its context to analyze it and form a concept about it.
Digital literacy	The ability to use digital technology, communication tools, or networks to locate, evaluate, use, and create information.
Algorithm	Instructions given in a particular order.
Prediction	The ability to estimate the development of various events, the results of actions, or phenomena.
Programming area	Where the program is written in ScratchJr.
Automatic	A mechanism that works by itself.
Programming block	A single instruction for a ScratchJr character.
Loop	An instruction that repeats itself.
Computer Science	A scientific discipline that studies computers and algorithmic processes, including their principles, hardware and software designs, and implementation, as well as its impact on society.

Code	Code written in a programming language to tell a computer what to do.
Grid	Lines that intersect to form squares.
Debugging	Process of finding problems in a computer program and trying to solve them in different ways.
Scene	A part of a whole story.
Control structures	Pattern recognition and repetition, cause and effect.
Background	The part of an image that is behind the main object.
Hardware	The objects we use when using computers, such as the CPU, tablet, keyboard, or mouse.
Programming languages	Language used by computers.
Modularity	The ability to divide large tasks into smaller parts.

Glossary

Order	The way a list of things is arranged. Order matters in both human and computer languages.
Page	Where a book or a story is written or drawn.
Block palette	Where blocks are in ScratchJr.
Parameter	A numerical instruction that tells ScratchJr how many times to do something.
Pattern	A recognizable, repetitive, and predictable structure that can be a visual pattern, a series of events, a mathematical structure, or even a behavior, created to solve a situation or problem.
Computational problems	Solving problems using programming concepts and processes (such as conditional statements, loops, and variables).
Design process	Understanding the cyclical nature of creative processes and its six steps: perseverance, debugging, problem identification and resolution, developing strategies to make things work, and problem-solving.

Programmer	A person who writes instructions for computers to tell them what to do.
Representation	Symbolic representation, models.
ScratchJr	ScratchJr: a programming language with images.
Sequence of algorithms	Putting things in order, logical organization.
Sequence of programming commands	ScratchJr blocks interlocked together.
Software	The computer programs we use when working on computers, such as the ScratchJr application.

Ceibal

Pensamiento Computacional e Inteligencia Artificial