



# PIAF - Pedagogical Scenario

(PIAF = Développement de la pensée informatique et algorithmique dans l'enseignement fundamental – Development of computational and algorithmic thinking in basic education)

#### **Title**

Guide me in Codes

#### **Practical Information**

(Ideal) Number of students: 16 Age of the students: 9 – 12 years

Duration of the scenario: 3 sessions of 45 minutes each

#### Main discipline of the Scenario

C 2.5 Combining action sequences to achieve a goal

C 6.1 Verify if a sequence of actions reaches a given goal

C 6.2 Fix a sequence of actions for reaching a given goal

#### **Description**

Learners complete a series of exercises on a block programming interface involving a character and mazes. The exercises, of increasing complexity and difficulty, teach the learners about basic programming actions, creation of programming algorithms, and debugging existing code.

### PIAF-specific competencies/goals

# C1 Competency 2: Compose/decompose a sequence of actions > Combining action sequences to achieve a goal > Learners combine different blocks in a given order to allow the character to complete the maze. C2 Competency 6: Build a sequence of actions iteratively > Verify if a sequence of actions reaches a given goal > Learners test the algorithms they created and see on the output window the execution of their algorithm to verify is the target is reached C3 Competency 6: Build a sequence of actions iteratively > Fix a sequence of actions for reaching a given goal > Learners are given either a) a faulty algorithm that they need check and correct, b) incomplete code that they need to complete, or c) no code for which they need to create an algorithm fully by their own, and checking and correcting it when necessary













# Pre-requisite for the activities

- o Being able to read and follow instructions
- o Being able to work on a computer
- o Understand directions and numbers
- o Spatial ability

# **Digital Resources**

| Technical  | Didactic                     |
|--|------------------------------|
| Computers with a web browser; 1 computer per student plus 1 for the teacher Projector Digital access to the exercises (either by internet or by downloading the exercises to the computer) | Course notes and attachments |





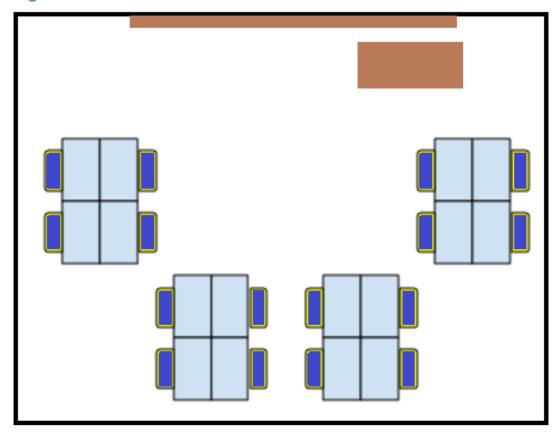








# Organization of the classroom













(5')



#### Scenario (Sequence of the activities)

| Activity 1: Introduction to the Blockly platform and Sequencing |              |                           |  |
|---|--------------|---------------------------|--|
| 1.  | Introduction | Group Format: Whole class |  |

Instruction: "Today we will learn about Edith. Edith is a student of your age and we will help her with her first day at a new school. For helping her, we will use our computers and program instructions."

Students task: Students interact by responding to questions Instructors' role: Introduce the task and answer students questions

2. Intro to Blockly platform (10') Group Format: Whole class

**Document:** Attachment A for teachers

Instruction: "Now that each of you have a computer, please start the program. You should be now seeing the 1st level. At the top, you have the instructions for this level. On the left side, you should see a map showing a character, a path and a tile with a target. That section is called "Viewer". That character is Edith and the tile with the target indicates where Edith needs to reach. On the middle of the screen, you can see 1 block that says "move forward". On that middle section you will have the blocks that you can use for each level. That section is called "Toolbox". On the right side you will find the "Editor" section. There you can see another "move forward" block, however that is the section where we will connect the blocks to create the instructions that will make Edith move and reach her goal. Try to click on the "move forward" block in the middle, drag it to the section on the right and connect it to the existing block. When you do that, you will hear a click. You can click the "Run" button on the left side to see what your code, the blocks you connected, does. If Edith didn't reach the school entry, press the "Reset" button, add or remove the necessary blocks and try again. If you want to delete a block, you can click on it and drag it to the bottom right corner where a trashcan is. Let's do the first level at the same time. When you're done, raise your hand." Students task: Listen to the teacher's instructions. Complete the first level

<u>Instructors' role</u>:. After explaining the Blockly interface, ensure all students complete the first level. Provide help if students struggle in exploring learning materials.















| ELASITIUST              |   |                   |
|-------------------------|---|-------------------|
|                         | Expected response: All students should be able to understand the layout of the Blockly interface and complete the first level.  |                   |
| 3. Exercises (20')      | Group Format: Individual Document: Attachment B for teachers Instruction: "You can now start with exercises. If you have any questions or troubles, just raise your hand and I will help you. Remember to read the instructions of each level at the top of the screen. Some levels might have different configuration. For example, a faulty code level is a level with error blocks. Your task would be to identify and correct any mistakes there. An incomplete code level is a level with missing blocks. To reach the goal, your task would be to identify and insert any missing blocks there.  Students task: Complete the exercises for the session Instructors' role: Provide support. Once the time is up, note the last completed level each student did. In the case of students struggle with either the faulty code level or the incomplete code level. Try guiding them to run a current code and see how far Edith could go. | 2.5<br>6.1<br>6.2 |
| 4. End of session (10') | Group Format: Whole class Instruction: "Let's review what we learned today" Students task: Verbal description of what has been learned during this session. Instructors' role: Guide the students with questions for obtaining the expected answers. Covered topics: -Blocks they learned and how to use them -What's the sequence? How could you help Edith reaching goals? -Could you give an example of sequencing in your real life? This could be a set of action you do in daily life (E.g. What's a sequence of action to travel from your home to the school?)  |                   |
| Activity 2: Loop co     | ding  |                   |
| 1. Reminder (5')        | Group Format: Whole class  Instruction: "Can someone remind me of what we did last time?"  Students task: Verbal description of the   |                   |













| Erasmus+                      | 100  |                   |
|-------------------------------|--|-------------------|
|                               | activities done in the previous session  Instructors' role: Guide the students with the things they learned last session  Covered topics students should mention: -Sequencing -Amount of exercises done last session -Blocks they learned and how to use them  |                   |
| 2. Introduction to loop (10') | Group Format: Whole class  Document: Attachment C for teachers  Instruction: "Today, I'm going to introduce you to Loop function. As you can see the exercise on your screen, you might notice that there's a new block called Repeat x times. The repeat block is used when you want to repeat a specific block. The blocks inside the repeat block will continuously repeat until they reach the exact amount of parameter on the repeat block. For example, if I insert one Move Forward block inside the repeat 2 times block, it will move forward twice. Now, you can try to solve the first exercise on your screen" Students task: Listen to the teacher's instructions. Ask questions if necessary. Instructors' role:. Explain and answer questions Expected response: All students should be able to understand how the "repeat" block works. |                   |
| 3. Exercises (20')            | Group Format: Individual  Document: Attachment C for teachers  Instruction: "You can now start and complete the rest of exercises. If you have any questions or troubles, just raise your hand and I will help you. Remember to read the instructions of each level at the top of the screen. In some levels, you might face another block called Repeat Until. This block works as same as the regular repeat block. But, it will keep repeating until reaching goals."  Students task: Complete the exercises for the session Instructors' role: Provide support. Once the time is up, note the last completed level each student did.   | 2.5<br>6.1<br>6.2 |
| 4. End of session (10')       | Group Format: Whole class Instruction: "Let's review what we learned today" Students task: Verbal description of what has been learned during this session Instructors' role: Guide the students with  |                   |















|                                    | questions for obtaining the expected answers Covered topics: -What's repeat and repeat until blocks? How to use them? -What's looping? Could you give an example of looping activities in your real life? (E.g. Systems like ATM machine, Music playlists that will repeat like a loop) |   |
|------------------------------------|---|---|
| Activity 3: Conditional expression |   | - |

| •                                     |   |                   |
|---------------------------------------|---|-------------------|
| 1. Reminder<br>(5')                   | Group Format: Whole class  Instruction: "Can someone remind me of what we did last time?"  Students task: Verbal description of the activities done in the previous session  Instructors' role: Guide the students with the things they learned last session.  Covered topics students should mention:  -What's repeat and repeat until blocks? How to use them? -Looping   |                   |
| 2. Introduction to conditionals (10') | Group Format: Whole class  Document: Attachment D for teachers  Instruction: "Today, we are going to learn about Conditionals.  Conditionals are expressions that can be evaluated to True or False. On your screen, you might notice that we have a new block called If Path. You need to first define a condition or parameter, and if that condition is true, the block or blocks inside it will be executed. For example, if there's a path ahead, Edith can move forward. This means that if there's a path ahead of Edith (True condition), Edith will move forward. In this exercise, you can combine the if path block with repeat blocks you learned last time. You might change the parameter on if path block by using drop down arrow on the if path block."  Students task: Listen to the teacher's instructions. Ask questions if necessary.  Instructors' role:. Explain and answer questions  Expected response: All students should be able to understand how the "If Path" block works. |                   |
| 3. Exercises (15')                    | Group Format: Individual  Document: Attachment D for teachers   | 2.5<br>6.1<br>6.2 |















| LIUSTIUST                          |  |  |
|------------------------------------|--|--|
|                                    | Instruction: "You can now start and complete the rest of exercises. If you have any questions or troubles, just raise your hand and I will help you. Remember to read the instructions of each level at the top of the screen."  Students task: Complete the exercises for the session Instructors' role: Provide support until students reach the final level of conditional exercises.   |  |
| 4. If-else<br>conditionals<br>(5') | Group Format: Whole class  Document: Attachment D for teachers Instruction: "In the last exercise, you might notice that there's a new block called If-else block. This block works almost the same as if path blocks. But, it will contain a result in case of having a false condition. For example, if path has a stone, pick up, else move forward. This means if there's a stone on the path, then Edith will pick it up. Otherwise, she will move forward."  Students task: Complete the exercises for the session Instructors' role: Provide support. Make sure that students understand the concept of the if-else block |  |
| 5. End of session (10')            | Group Format: Whole class Instruction: "Let's review what we learned today" Students task: Verbal description of what has been learned during this session Instructors' role: Guide the students with questions for obtaining the expected answers Covered topics: -What's if path and if-else blocks? How do we use them? What's the differences between those two types of blocks? -What's conditionals? Could you give an example of conditionals in our real life? (E.g. If I get more than 50 points out of 100 points in the exam, I will pass. Or, it could be any kind of condition in the world)                        |  |













#### **Assessment**

Learners will complete a final set of exercises that combine the functions and blocks that they have learned.

| Competencies/ PIAF-<br>Goals                               | Activities for the assessment           | Assessment criteria   |
|--|---|---|
| C 2.5 Combining action sequences to achieve a goal         | Use coding and variables                | Help the character reach its goal   |
| C 6.1 Verify if a sequence of actions reaches a given goal | Run the written codes                   | Reaching the goal   |
| C 6.2 Fix a sequence of actions for reaching a given goal  | Debugging and rewriting algorithm codes | Code and target are presented to students. Students should find any present 'bug' and edit the algorithm. |

#### Received Feedback on the created Scenario

If you have had the opportunity to experiment with the scenario presented here, suggest some feedback on it: what worked well, the obstacles encountered, the learner's feedback, your feelings, possible ways to improve it.













Image assets: Created by Delsin53, available here: <a href="https://delsin53.itch.io/pixel-tileset">https://delsin53.itch.io/pixel-tileset</a>

Blockly library: Blockly is a library that adds a visual code editor to web and mobile apps. Blockly is an open source project supported by Google. For more information, please check: <a href="https://developers.google.com/blockly/guides/overview">https://developers.google.com/blockly/guides/overview</a>













#### **Attachments Overview**

| Activity | Teacher Attachment | Student Attachment |
|----------|--------------------|--------------------|
| 1.2      | Α                  |                    |
| 1.3      | В                  |                    |
| 2.2      | С                  |                    |
| 2.3      | С                  |                    |
| 3.2      | D                  |                    |
| 3.3      | D                  |                    |
| 3.4      | D                  |                    |







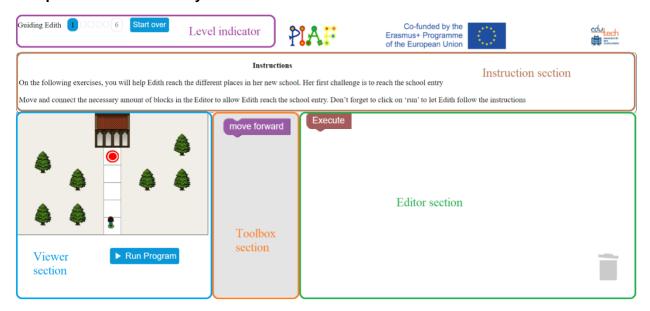




#### Teacher's Attachment: A

| Used in activity:                   | 1.2: Intro to Blockly platform |
|-------------------------------------|--------------------------------|
| Along with Student's Attachment(s): | none                           |

#### Components of the Blockly interface



- The completion progress is stored in the web browser. On the level indicator, the button "Start over" will delete any completed levels. This information cannot be restored.
- Levels that have been completed will appear as white square in the Level indicator section. A blue square indicates the current stage or the stage that have already done.
   In the image above, levels 1 to 6 have been completed.





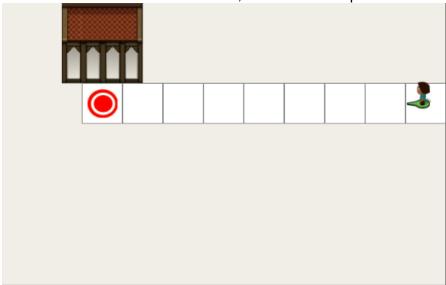








• On further levels, only a limited amount of blocks can be added in the Editor section. This limit will be shown in the Viewer section, between the map and the "Run" button:





You have 2 blocks left.







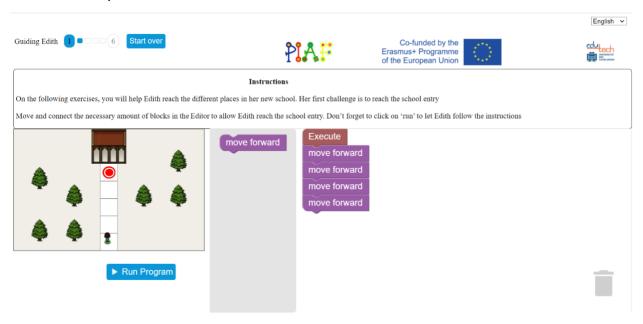








#### Solution for Sequence exercise: 1.1













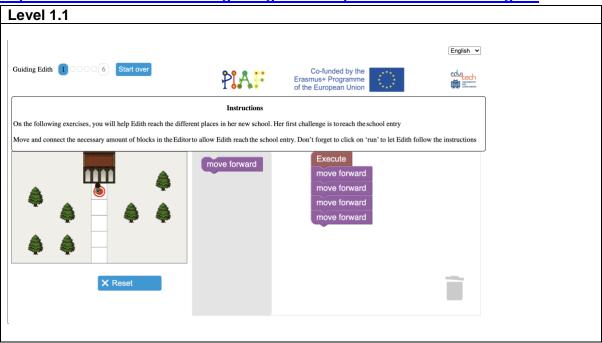


#### Teacher's Attachment: B

| Used in activity:                   | 1.3: Exercises |
|-------------------------------------|----------------|
| Along with Student's Attachment(s): | none           |

#### First level set - Sequences

https://edutech.uni-saarland.de/guidingedith2/sequences/maze.html?lang=en







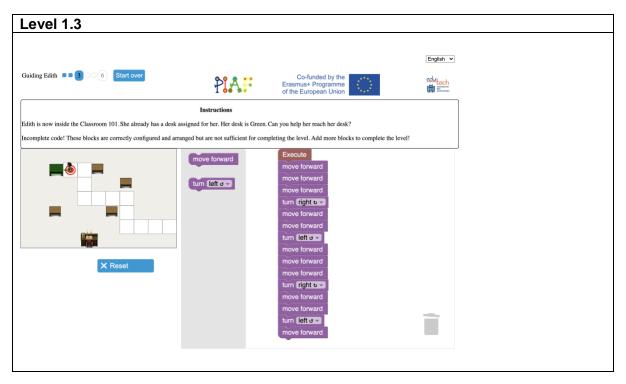


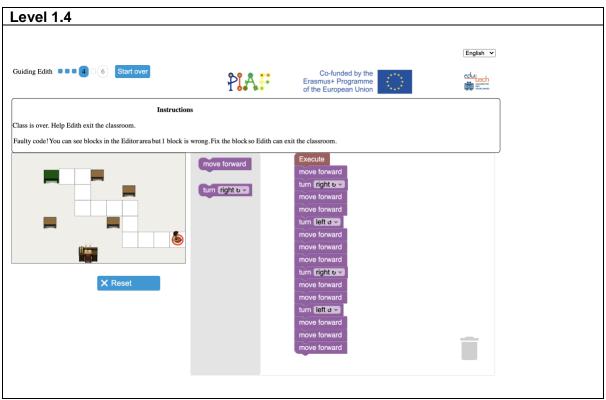














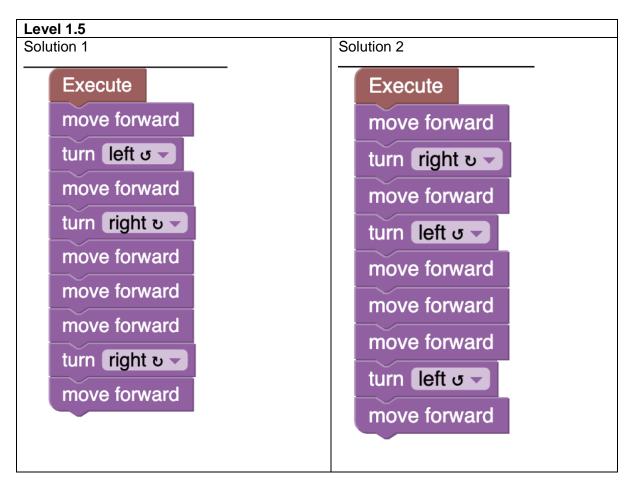


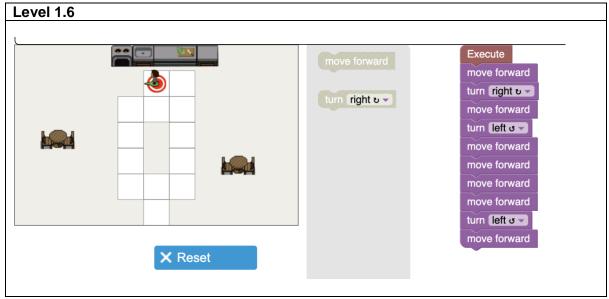
























#### Teacher's Attachment: C

| Used in activity:                   | 2.2: Introduction to loop |
|-------------------------------------|---------------------------|
|                                     | 2.3: Exercises            |
| Along with Student's Attachment(s): | none                      |

#### Second level set - Loops:

https://edutech.uni-saarland.de/guidingedith2/loops/maze.html?lang=en







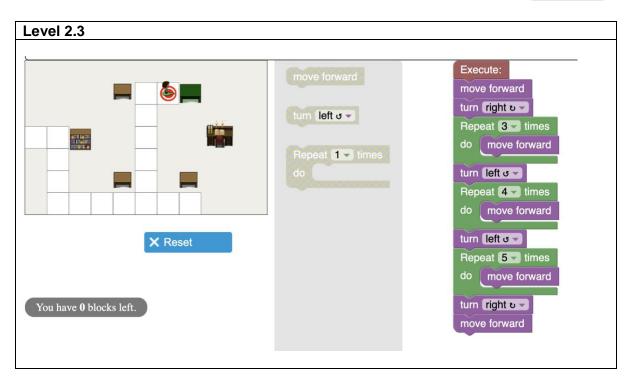


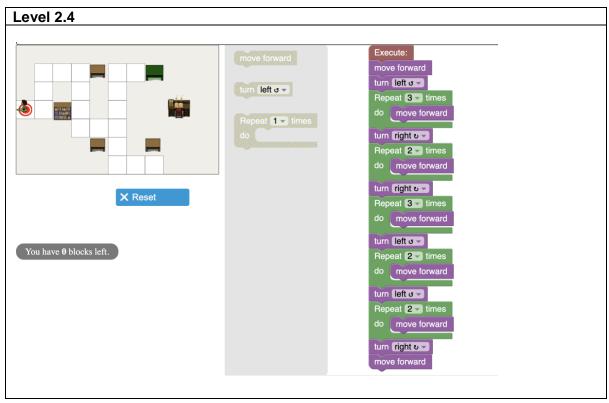














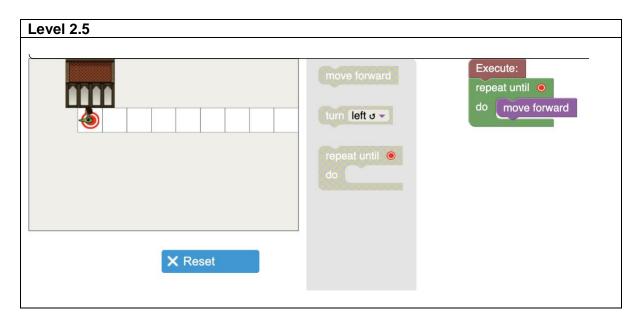


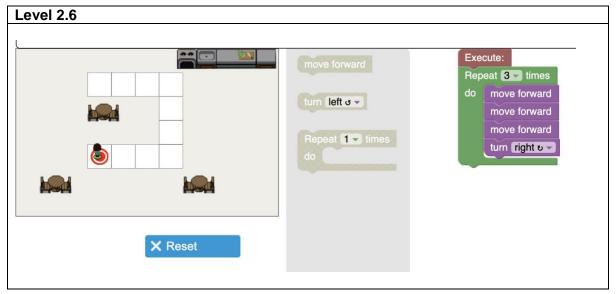
























# Level 2.7 Execute: Repeat 4 times do move forward turn left o turn left o move forward Repeat 1 times turn right อ 🗸 X Reset











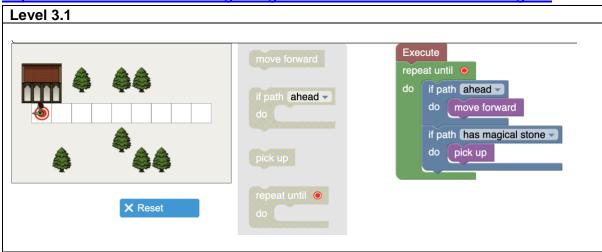


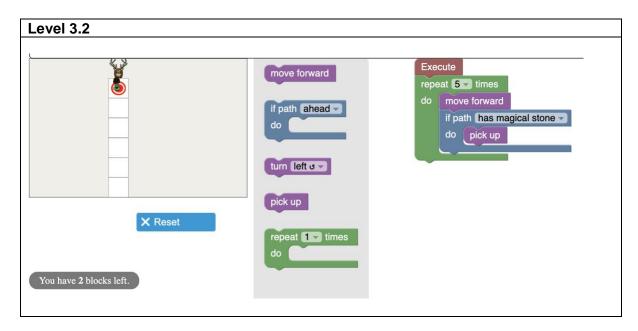
#### Teacher's Attachment: D

| Used in activity:                   | 3.2 Introduction to conditionals |
|-------------------------------------|----------------------------------|
|                                     | 3.3 Exercises                    |
|                                     | 3.4 If-else conditionals         |
| Along with Student's Attachment(s): | none                             |

#### Third level set - Conditionals

https://edutech.uni-saarland.de/guidingedith2/conditionals/maze.html?lang=en







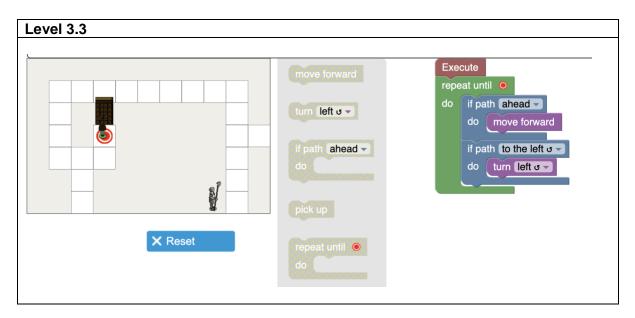


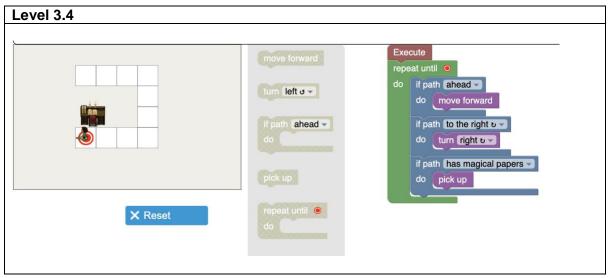












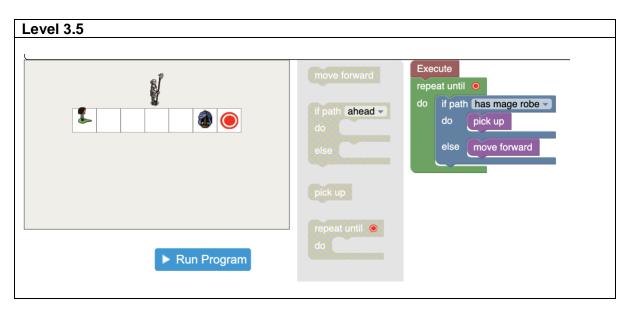






















The evaluation activity is the combination of what students have learned from 3 sessions (Sequencing, Loop and Conditionals). There are 4 exercises in total. Because these exercises are quite challenging, teachers might provide additional guide on how "Nested Loop" works. Teachers might clarify that "Repeat blocks can be placed inside another repeat blocks"

#### https://edutech.uni-saarland.de/guidingedith2/nestedloops/maze.html?lang=en

