Trainers Support

Algorithmic Thinking

Algorithmic Thinking for Migrants Teachers Education

2021-1-EL01-KA210-ADU-000035033

LESSON #4

TITLE: PATTERNS & GENERALIZATION

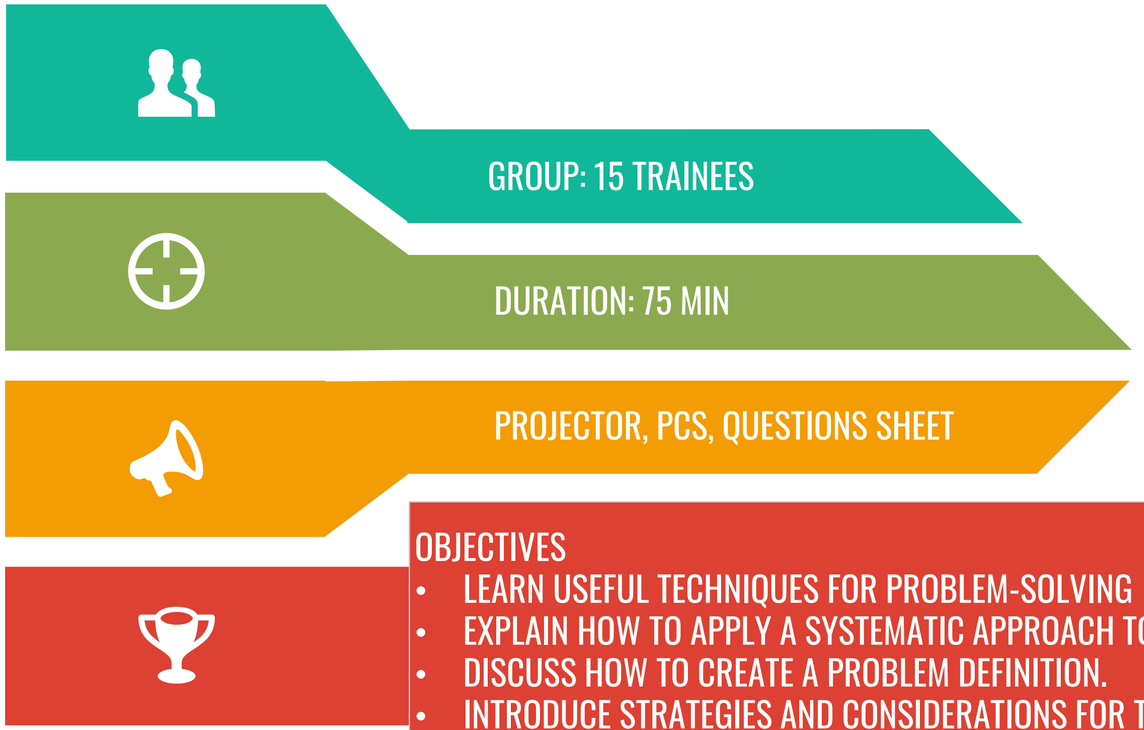
ΙΔΡΥΜΑ ΚΡΑΤΙΚΩΝ ΥΠΟΤΡΟΦΙΩΝ IKY

Co-funded by the Erasmus+ Programme of the European Union

LESSON #4 – PATTERNS & GENERALIZATION

Erasmus+
Ανοίγει δρόμους, αλλάζει ζωές.

LESSON REQUIREMENTS



OBJECTIVES

- LEARN USEFUL TECHNIQUES FOR PROBLEM-SOLVING
- EXPLAIN HOW TO APPLY A SYSTEMATIC APPROACH TO PROBLEM-SOLVING.
- DISCUSS HOW TO CREATE A PROBLEM DEFINITION.
- INTRODUCE STRATEGIES AND CONSIDERATIONS FOR THE DEVISING OF SOLUTIONS.
- EXPLAIN DECOMPOSITION AS A PROBLEM-SOLVING STRATEGY.

ΙΔΡΥΜΑ ΚΡΑΤΙΚΩΝ ΥΠΟΤΡΟΦΙΩΝ IKY
Co-funded by the Erasmus+ Programme of the European Union

LESSON #4 – PATTERNS & GENERALIZATION

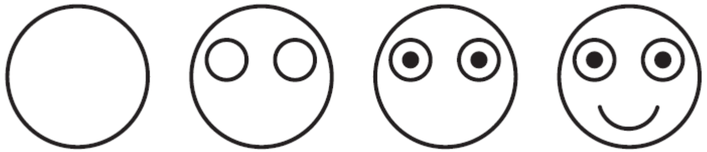
WHY LOOK FOR PATTERNS ???

By combining this strategy with **decomposition**, we might find we could draw a smiley face by breaking it down into several simpler shapes, which are each easy to draw:

1. A circle, to serve as the head.
2. Two concentric circles (the smaller one of which is solid) which make eyes.
3. A curve, which makes the mouth.

Smiley face.

Breakdown of drawing smiley face.

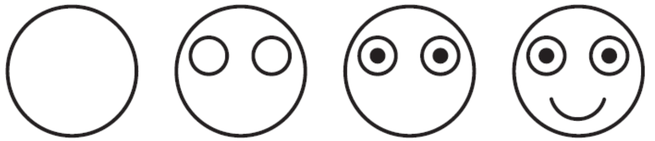


LESSON #4 – PATTERNS & GENERALIZATION

WHY LOOK FOR PATTERNS ???

An algorithm that draws such an image might therefore look like this:

1. Draw circle with radius 30 at position 50,50 with line thickness
2. Draw circle with radius 6 at position 40,40 with line thickness 1.
3. Draw circle with radius 3 at position 40,40 filled black.
4. Draw circle with radius 6 at position 60,40 with line thickness 1.
5. Draw circle with radius 3 at position 60,40 filled black.
6. Draw red line from position 30,70 to position 70,70 with line thickness 1.



Breakdown of drawing smiley face.

LESSON #4 – PATTERNS & GENERALIZATION

RECOGNIZING SIMPLE PATTERNS

Here's one approach to spotting simple patterns:

- Look for **nouns** that appear repeatedly. These could correspond to objects that your solution deals with.
- Look for **verbs** that appear repeatedly. These could be operations that the solution carries out.
- Look for concrete descriptions. These could probably be substituted by placeholders that vary in different situations.

For example:

- **adjectives** ('red', 'long', 'smooth') which indicate properties of things and could be replaced by the property name (color, size, texture);
- actual **numbers**, which could be replaced with variables.

ΙΔΡΥΜΑ ΚΡΑΤΙΚΩΝ ΥΠΟΤΡΟΦΙΩΝ IKY
Co-funded by the Erasmus+ Programme of the European Union

LESSON #4 – PATTERNS & GENERALIZATION

Erasmus+
Ανοίγει δρόμους, αλλάζει ζωές.

PROPOSITIONS

The generalizations that have been extracted can be seen beside:

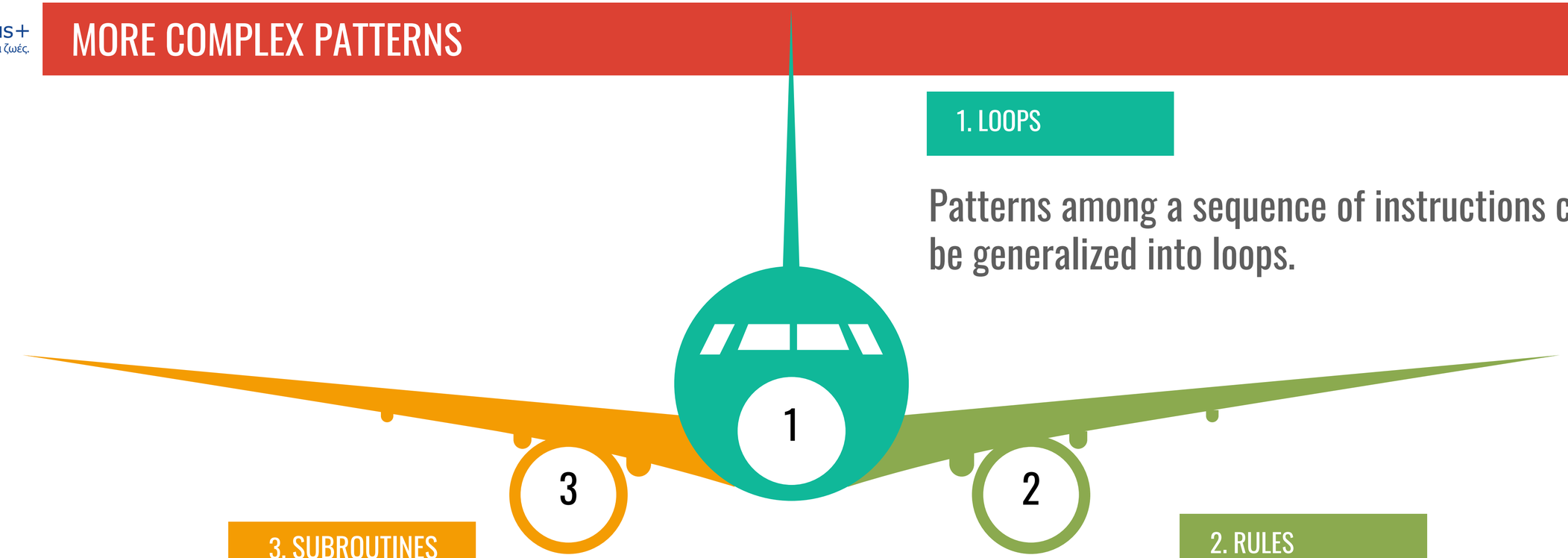
Properties

1. **Draw** (Shape): an operation that renders a shape.
2. **Fill** (Shape, Color): an operation that makes the interior of a shape solid with a color.
3. **Shape**: a form with an external boundary. Specific shapes include: (a) **Circle** (a type of shape with a radius, position and a line thickness) and (b) **Line** (a type of shape with two positions and a line thickness).
4. **Color**: red, blue, green and so on.

ΙΔΡΥΜΑ ΚΡΑΤΙΚΩΝ ΥΠΟΤΡΟΦΙΩΝ IKY
Co-funded by the Erasmus+ Programme of the European Union

LESSON #4 – PATTERNS & GENERALIZATION

MORE COMPLEX PATTERNS



1. LOOPS

Patterns among a sequence of instructions can be generalized into loops.

2. RULES

Patterns among conditionals or equations can be generalized into rules.

3. SUBROUTINES

Patterns among separate groups of instructions can be generalized into subroutines.

ΙΔΡΥΜΑ ΚΡΑΤΙΚΩΝ ΥΠΟΤΡΟΦΙΩΝ IKY
Co-funded by the Erasmus+ Programme of the European Union

LESSON #4 – PATTERNS & GENERALIZATION

Erasmus+
Ανοίγει δρόμους, αλλάζει ζωές.

LOOPS

Consider this block:

- Draw circle with radius 6 at position 40,40
- Draw circle with radius 3 at position 40,40 filled black
- Draw circle with radius 6 at position 60,40
- Draw circle with radius 3 at position 60,40 filled black

We see a pattern here. Together, the last two steps essentially repeat the first two, only with a slight difference in positioning.
We could therefore put these steps into a loop.

- Let coordinates = (40,40), (60,40)
- For each coordinate x,y do the following:
  1. Draw circle with radius 6 at position x,y
  2. Draw circle with radius 3 at position x,y filled black

ΙΔΡΥΜΑ ΚΡΑΤΙΚΩΝ ΥΠΟΤΡΟΦΙΩΝ
IKY
Co-funded by the Erasmus+ Programme of the European Union

LESSON #4 – PATTERNS & GENERALIZATION

SUBROUTINES

A generalized version might therefore look like this:

- Draw circle with radius r1 at position x,y
- Draw circle with radius r2 at position x,y filled black

We could then declare that these steps constitute a subroutine, that is to say, a sequence of instructions that perform a distinct, often-invoked task and which can therefore be ‘packaged’ as a unit. It would look something like this:

- ‘Draw eye’ is a subroutine (r1, r2, x,y):
  - Draw circle with radius r1 at position x,y
  - Draw circle with radius r2 at position x,y filled black

LESSON #4 – PATTERNS & GENERALIZATION

RULES

The drawing routines defined in the previous slide contain a pattern we've not yet picked out, although we might have spotted it already. When drawing an eye, the radius of the inner circle **is always half the radius of the outer circle**. If that is indeed the rule, then we can encode it explicitly into the subroutine so it becomes:

1. 'Draw eye' is a subroutine (r, x, y):
    2. Draw circle with radius r at position x,y
    3. Draw circle with radius ½r at position x,y filled black

LESSON #4 – PATTERNS & GENERALIZATION

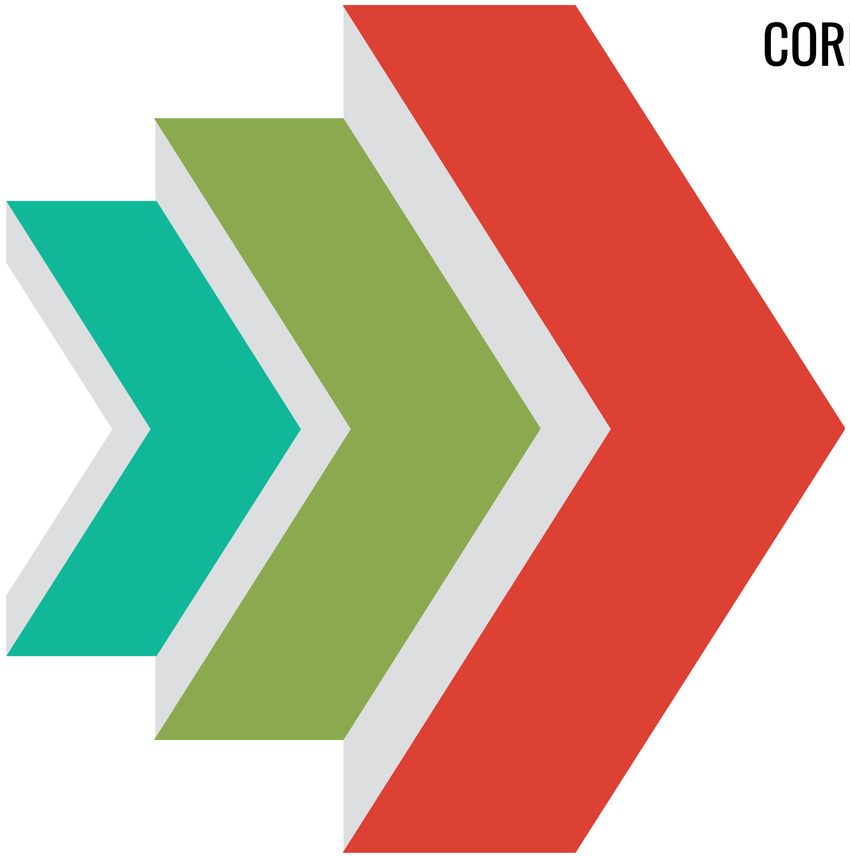
Erasmus+
Ανοίγει δρόμους, αλλάζει ζωές.

ACTIVITY #4.1

Trainer shares a sheet with two exercises.

Trainees, write down the answers for the next 20 minutes and at the end they all together discuss the results.

The discussion follows last 10 minutes.



CORE SKILLS DEVELOPED

- Problem solving skills
- Reasoning skills
- Generalization skills
- Technical skills
- Time management skills

TIMING

30 min

REQUIRED TOOLS

PC, projector, sheet

ΙΔΡΥΜΑ ΚΡΑΤΙΚΩΝ ΥΠΟΤΡΟΦΙΩΝ IKY
Co-funded by the Erasmus+ Programme of the European Union

REFERENCES

PANE, J. F. ET AL. (2001) STUDYING THE LANGUAGE AND STRUCTURE IN NON-PROGRAMMER'S SOLUTIONS TO PROGRAMMING PROBLEMS. *INTERNATIONAL JOURNAL OF HUMAN-COMPUTER STUDIES*, 54 (2). 237.

BEECHER, KARL. 2017. COMPUTATIONAL THINKING: A BEGINNER'S GUIDE TO PROBLEM-SOLVING AND PROGRAMMING. SWINDON, ENGLAND: BCS: THE CHARTERED INSTITUTE FOR IT.

ΙΔΡΥΜΑ ΚΡΑΤΙΚΩΝ ΥΠΟΤΡΟΦΙΩΝ IKY
Co-funded by the Erasmus+ Programme of the European Union