

Lesson Title

A Fun Introduction to Programming in Turing

Lesson Overview

Students will learn basic commands used for programming graphics in the Turing programming language, and will create a programmed landscape as the culminating project.

Learning Goals

- I will be able to use programming commands such as `drawbox()`, `drawline()`, and `drawoval()` to create primitive graphics shapes.
- I will be able to write short programs according to a set of instructions.
- I will help create the requirements for a landscape programming assignment.
- I will be able to write a program using a series of graphics commands in Turing to create a landscape.

Expectations

- A1.3 apply correctly the mathematical and scientific concepts and skills required in the planning and development of a product and/or service;
- A1.4 incorporate appropriate technological concepts (e.g., aesthetics, control, environmental sustainability/stewardship, ergonomics, fabrication/building/ creation, function, innovation, material, mechanism, power and energy, safety, structure, systems) in the design, fabrication or delivery, and evaluation of a product or service (see pp. 5-6);
- A3.1 evaluate a product or service, and processes associated with its development, on the basis of a set of criteria relevant to that product or service (e.g., adherence to specifications, ease of use, attractive appearance, ruggedness, clean joints, acceptable weld bead, uniform colour, adherence to forest management plan, nutritional value);
- B1.1 apply the steps of a design process or other problem-solving process to plan and develop products and services (e.g., define the problem or challenge, taking into account relevant contextual or background information; gather information [about criteria, materials, constraints]; generate possible solutions, using techniques such as brainstorming; choose the best solution; develop and produce a model or prototype; test the model or prototype; incorporate improvements or redesign and retest; report on results) (see pp. 16-19);
- B2.1 use appropriate tools, materials, and equipment (e.g., tools: hammer, chisel, screwdrivers, soldering iron, cheese grater, sieve, seam ripper; pruning shears, hair clipper; materials: wood, aluminum, polystyrene, paper, wax, clay, textiles, electronic components, mulch, hair colour; equipment: drill press, test meter, computer, software, printer, video camera, thermometer, grill, sewing machine, autoclave, curling iron) to create products or deliver services;

Resources

- Turing Graphics Assignments
<http://www2.beens.org/ics/assign/turing-graphics>
- Turing Graphics Screen Coordinates Slideshow
<http://goo.gl/XIQn7K>
- Turing Graphics Commands (list)
<http://www2.beens.org/ics/turing/references/graphics-summary>
- Exemplars
<http://www2.beens.org/ics/assign/turing-graphics/example-landscapes>
- Previous class' summative brainstormed requirements
<http://goo.gl/p9s9O4>

Prior Knowledge

No prior knowledge is assumed.

Accommodations

- Student may elect to do this assignment in groups of two.
- For students that have trouble multitasking, the instructor can provide examples in the students' code to help them with the programming commands.
- For more advanced students, encourage them to research how to do the same assignment in Processing or Python (using Codeskulptor).

Minds on

Previous students' work can be shown from

<http://www2.beens.org/ics/assign/turing-graphics/example-landscapes>.

As this may be the first time many students have programmed, it is important to emphasize that the computer does exactly as it is told, in exactly the order it is told.

Action

1. Provide the students of an overview of the assignment.
2. Discuss the concept of programming, and give examples of typical program.
3. Explain the history of Turing and why it is a suitable language for learning how to program.
4. Show the students the exemplars, which is what they will be creating for their culminating activity.
5. Show the students the screen coordinates slideshow.

6. Show the students the graphics commands, highlighting the ones they will most likely be using.
7. Show the students the assignments page, and walk them through the first few assignments. Stress the importance of naming the files correctly and saving them to the correct place.
8. After the students have made some progress and a few of the students are getting near the landscape assignment, have a discussion about the landscape requirements and show them some previous brainstormed requirements as an example.
9. Brainstorm with the class the requirements for this class' landscape and document them in the shared Google Doc.
10. Give the students adequate time to finish the assignments.

Consolidation

- Do a “show-and-tell” of the students' landscapes.
- Ask the students what can be improved in this unit for future students.