

## ELTE

|  |                |                 |                     |
|--|----------------|-----------------|---------------------|
| <b>Name of course:</b> Scratch Programming   |                |                 |                     |
| <b>Course code:</b> IP-18fPRESPROGEG   |                |                 |                     |
| <b>Faculty responsible for course:</b> Márta Turcsányi-Szabó   |                |                 |                     |
| <b>Total credits:</b> 4  |                |                 |                     |
| <b>Total hours:</b> 4  |                |                 |                     |
| <b>Type of the course</b>  | <b>Lecture</b> | <b>Practice</b> | <b>Consultation</b> |
| Credit   | —              | 4               | —                   |
| Hours per week   | —              | 3               | 1                   |
| Type of testing  | —              | practice        | —                   |
| <p><b>Topics:</b></p> <p><b>Block programming, design-based learning introduction:</b><br/>Introduction to creative computing and Scratch using simple projects and hands-on experiences.</p> <p><b>Scratch Programming:</b></p> <ul style="list-style-type: none"> <li>- Introducing the Scratch user interface.</li> <li>- Exploring the arts by creating projects with sounds, music, drawings, and video.</li> <li>- Creating geometric figures using visual problem solving and turtle geometry.</li> <li>- Telling stories through animation projects featuring characters, scenes, and dialogues (simple interactive animation, single scene animation, multi-scene animation, endless animation).</li> <li>- Programming several kinds of games (adventure game, grid game, platformer game) with different goals and rules.</li> </ul> <p><b>3D modelling:</b><br/>Using the Tinkercad block programming and 3D design platform, and exporting the model for later use (e.g. character in a game)</p> <p><b>Assessment:</b> Students are evaluated by the number of points they gain during the semester. Points can be earned for:</p> <ul style="list-style-type: none"> <li>- Quizzes during the lesson</li> <li>- Small assignments (either completed at the end of the lesson or as homework)</li> <li>- Final project (creating a game by following a few guiding points)</li> </ul> <p><b>Competencies:</b><br/>Enabling user friendly programming environment using Scratch. Introducing learners to <i>creative computing</i> with Scratch, using a <i>design-based learning</i> approach. Providing the learners with opportunities to explore <i>computational thinking concepts</i> (sequence, loops, parallelism, events, conditionals, operators, data), <i>practices</i> (working iteratively and incrementally, testing and debugging, reusing and remixing, abstracting and modularizing), and <i>perspectives</i> (expressing, connecting, questioning, collaboration). Transferring of concepts learnt in different areas through development of projects.</p> |                |                 |                     |

**Literature:**

1. Digital Technologies  
<https://fuse.education.vic.gov.au/Resource/LandingPage?ObjectId=c1d4617e-a3c5-4131-937c-309de3ea4899>
2. How to use Scratch for Digital Story Telling  
<https://www.common sense.org/education/blog/how-to-use-scratch-for-digital-storytelling>
3. Story Telling with Scratch  
<https://scratch.mit.edu/projects/96741560/>
4. Scratch overview video  
<http://vimeo.com/29457909>
5. Scratch Tutorials and Basic Games
6. <https://scratch.mit.edu/studios/3457447/>
7. Dance videos  
<http://vimeo.com/28612347>  
<http://vimeo.com/28612585>  
<http://vimeo.com/28612800>  
<http://vimeo.com/28612970>
8. *Maze extension* projects:  
<http://scratch.mit.edu/galleries/view/138300>
9. Scratch Programming Video Tutorials  
<https://scratch.mit.edu/help/videos/>

**Recommended literature:**

1. Getting Started with Scratch  
<https://resources.scratch.mit.edu/www/guides/en/Getting-Started-Guide-Scratch2.pdf>