



## REGULATIONS

**Due date:** 28 November 2021, Sunday, 23:59 (*Not subject to postpone*)

**Submission:** You should compress all your Scratch solutions into a file named as `Scratch_tasks.tar.gz` and submit this compressed file at the ODTUClass page of the course.

**Team:** There is **no** teaming up. The homework has to be done and turned in individually.

**Cheating:** Source(s) and Receiver(s) will receive zero and be subject to disciplinary action.

## 1 Introduction

In this task we will ask you to solve some simple tasks with Scratch. You can consider these tasks as an entrance point to programming world. You will learn how to use Scratch, but beyond that, you will become familiarized with some crucial concepts of programming.

### What is Scratch?

Scratch is a high-level visual programming language, where you can develop games, animations etc. For more information: <https://scratch.mit.edu/about>

Scratch is widely used around the world. In *this link*, you can find many tutorials on how to use it, or how you can develop new codes.

There are also many projects that you can analyze. See e.g. *this link* for some example projects. You are highly recommended to experiment with different programs. You can see the codes, however it would be beneficial for you to try to implement the games or animations yourself.

### How to Use Scratch

You can code in Scratch online through the *web page*, or you can download the Scratch environment from its *official web page*.

When you start coding, blocks will be available on the left panel. In the middle panel, you can write your scripts (by dragging and dropping the code blocks). Finally on the rightmost panel, you can see the stage, which will include the default sprite of Scratch (🐱) when you open a new project.

After you finish coding, you can click the start button (🚩) to run your script.

## 2 Programming Tasks

These tasks will be graded as a ‘Participation’ activity in the grading plan announced in the syllabus.

### Task 1

In the first task, you should make the sprite move. When the script starts (  is pressed) sprite should move 10 steps.

**Bonus:** In this task, there are several other features of sprites that you can select. You can select one of the following:

1. You can change the default sprite with any other object.
2. You can change the background.
3. You can change the number of steps.
4. You can add a sound effect after the step.

Save your solution from the File menu with the name `task1.sb3`.

### Task 2

In the first task, sprite only moved 10 steps. In this task, it should do the following;

- When the script starts,

Step 1: Sprite moves 2 steps

Step 2: It waits for 1 second

Step 3: Sprite moves 2 steps


Step 4: It waits for 1 second

Step 5: Sprite moves 2 steps

At the end, Sprite should have moved 6 steps. Save your solution from the File menu with the name `task2.sb3`.

### Task 3

In the first two tasks, the number of steps Sprite made was limited and we explicitly wrote the steps repeatedly. In this task, Sprite will move 2 steps *forever*. In order to complete this task, you can use the *forever block* (under the Control menu).

Notice that Sprite will continue to move until you press stop button (  ).

Save your solution from the File menu with the name `task3.sb3`.

### Task 4

As you might have noticed, Sprite leaves the screen in Task 3, since it moves along one direction. In this task, you will make Sprite turn back, when it hits the wall. Similar to Task 3, Sprite should make two steps forever.

Save your solution from the File menu with the name `task4.sb3`.