MIDDLE EAST TECHNICAL UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING

CENG 111

Fall 2022

Scratch Assignment

REGULATIONS

Due date: 30 October 2022, Sunday, 23:59 *(Not subject to postpone)*

Submission: You should compress all your Scratch solutions into a file named as `Scratch_tasks.zip` 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. These are some warm-up tasks to get you thinking about certain programming constructs.

What is Scratch?

Scratch is a high-level visual programming language, where you can program by bringing together programming constructs as these constructs are the pieces of a jigsaw puzzle. For more information, see: *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 more beneficial for you to try to implement the games or animations yourself.

How to Use Scratch

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

When you start coding, blocks (pieces of the jigsaw puzzle) 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. Before starting, you have to change the default sprite ( ) to a baseball ( ). Right click to the sprite on the bottom-right panel and delete it. After that, click to choose a sprite button ( ) and select the baseball ( ). Then double left click on the baseball sprite on the bottom-right panel to make sure that you are coding that sprite.

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, your script should do the following;

- When the script starts:

Step 1: The sprite moves 3 steps.

Step 2: It waits for 1 second.

Step 3: The sprite moves 1 steps.

Step 4: It waits for 1 second.

Step 5: The sprite moves 3 steps.

At the end, the sprite should have moved 7 steps.
Save your solution into a file named `task1.sb3` using the File menu.

Task 2

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

Notice that the sprite will continue to move until you press stop button ( ).
Save your solution into a file named `task2.sb3` using the File menu.

Task 3

As you might have noticed, the sprite leaves the screen in Task 3, since it moves along one direction. In this task, you will make the sprite turn back, when it hits the wall. Similar to Task 2, the sprite should make two steps forever.

Save your solution into a file named `task3.sb3` using the File menu.

Task 4

Observe that, in Task 3, the sprite just bounces back and forth. In this task, you have to orient the sprite to a **random degree between 5 and 60** clockwise before it starts to move. Note that an orientation with **0** degree points **upward** in scratch. Note that you have to use *pick random* block (under the Operators menu) for picking random numbers for orientation. When the sprite hits a wall, it needs to bounce back with a direction symmetrical to the surface normal (just like a light ray would reflect from a reflective surface).

Save your solution into a file named `task4.sb3` using the File menu.