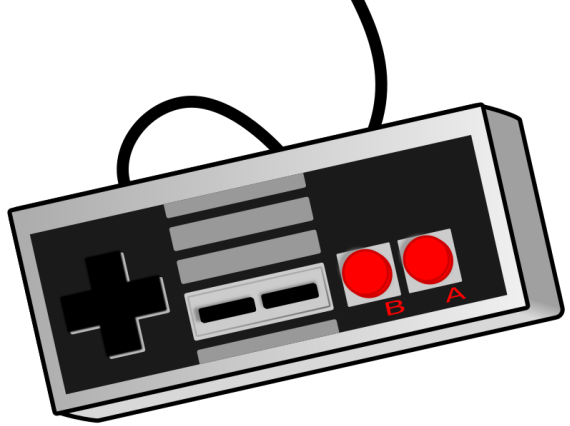


Lab_6

Playing Music

Outline

- Setting up the Crab Scenario
- Playing Music in Greenfoot
- Finish coding the Lobster
- Finding music online

Submission

You must create **Lab6_Answer** folder, where you will include:

1. **Greenfoot folder** of your completed game
2. **Lab6_Report.docx** file (which shows screenshots of the added code and execution view for each step..

Step 1: Setting up the Crab Scenario

The musical-crab Scenario

- Download the **Lab_6.zip** file from Omnivox, which contains the **musical-crab** Scenario
- Unzip the contents to somewhere on your USB key or hard disk.
- Open the scenario in that location with **Greenfoot**
- You should see the standard Greenfoot interface with the corresponding world, as shown in Figure 1

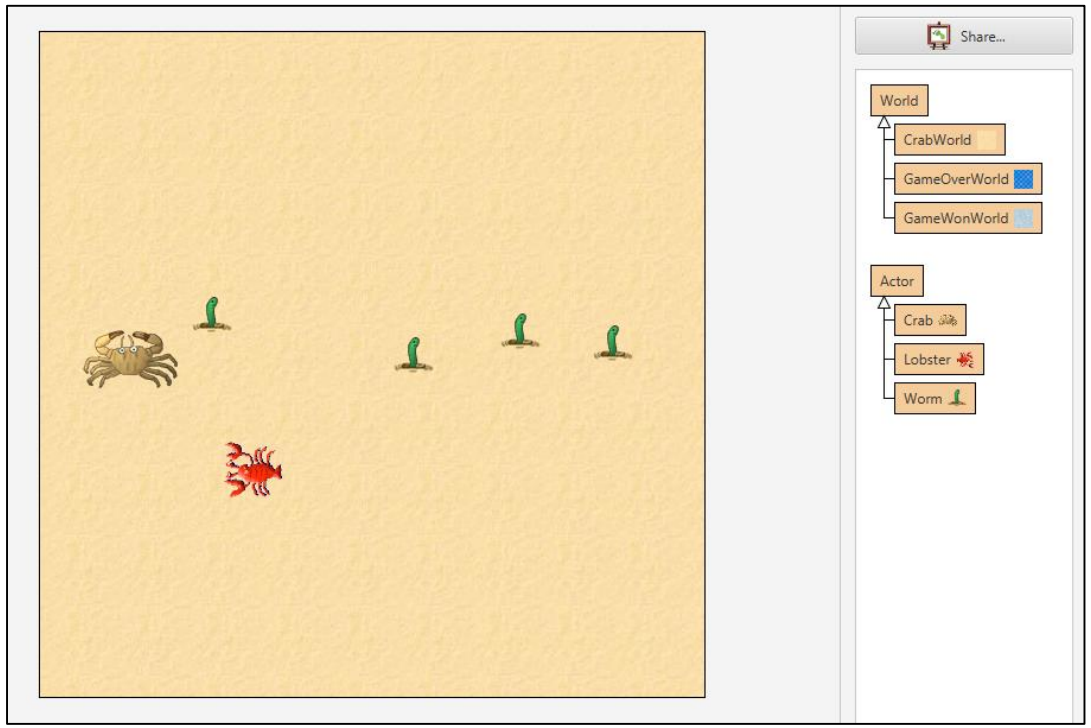


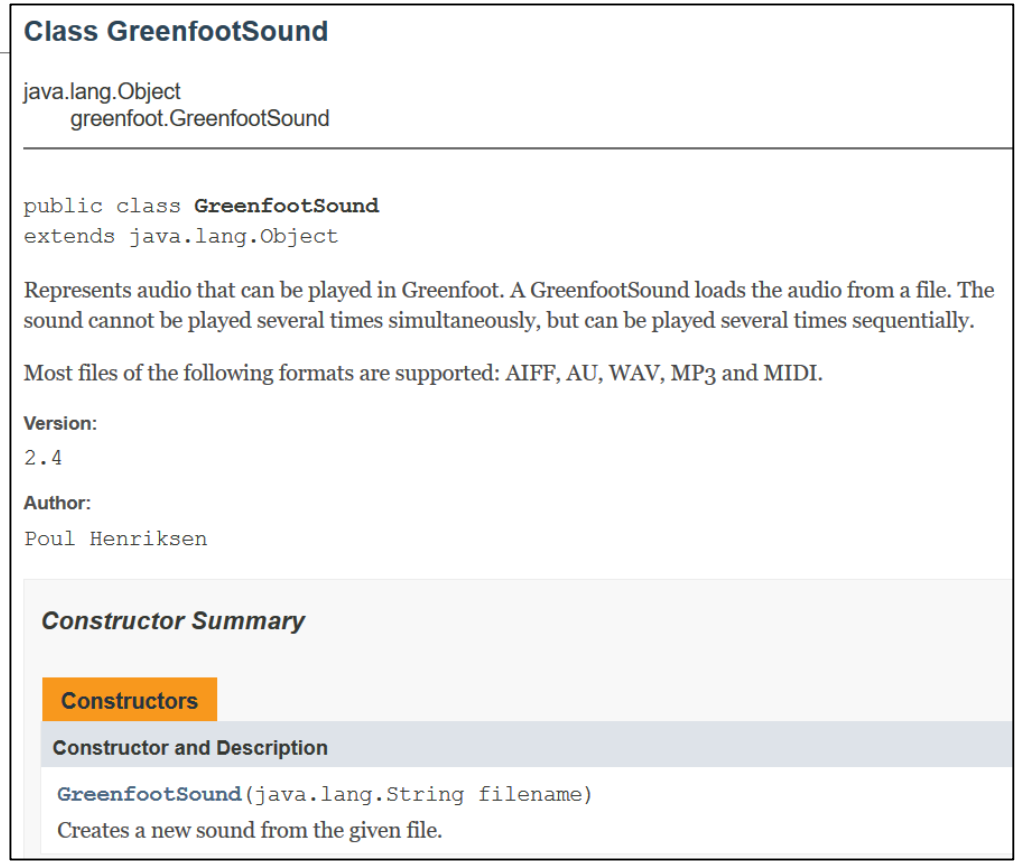
Figure 1

Step 2: Playing Music in Greenfoot

Adding music to your games increases the fun factor!

On specific events, changing the music adds to the game experience

- When being invincible in ***Super Mario***, the music speeds up, and back to normal when music invincibility wears off
- **Beatsync** is an important feature for professional games, it allows seamless transition between music tracks.
- More on this later…



Class GreenfootSound

java.lang.Object
greenfoot.GreenfootSound

```
public class GreenfootSound
extends java.lang.Object
```

Represents audio that can be played in Greenfoot. A GreenfootSound loads the audio from a file. The sound cannot be played several times simultaneously, but can be played several times sequentially.

Most files of the following formats are supported: AIFF, AU, WAV, MP3 and MIDI.

Version:

2.4

Author:

Poul Henriksen

Constructor Summary

Constructors

Constructor and Description
`GreenfootSound(java.lang.String filename)` Creates a new sound from the given file.

Figure 2

Step 3

In Greenfoot World, the **started()** and **stopped()** methods are automatically invoked when the scenario is **run** or **paused**.

The example on the right is playing back music on a loop

a. Create a ***GreenfootSound*** member variable
b. Instantiate the ***Greenfoot sound*** in the constructor
c. Start the playback when the scenario is started
d. Stop the playback when the scenario is paused or stopped

Figure 3

Step 4
Transition between worlds

a) For transition between worlds (from **CrabWorld** to **GameWonWorld**): we need to invoke the **started()** and **stopped()** method by code (see Figure 4a on the right):

- invoke the stopped() method in the ***CrabWorld***
- invoke the started() method in the ***GameWonWorld***

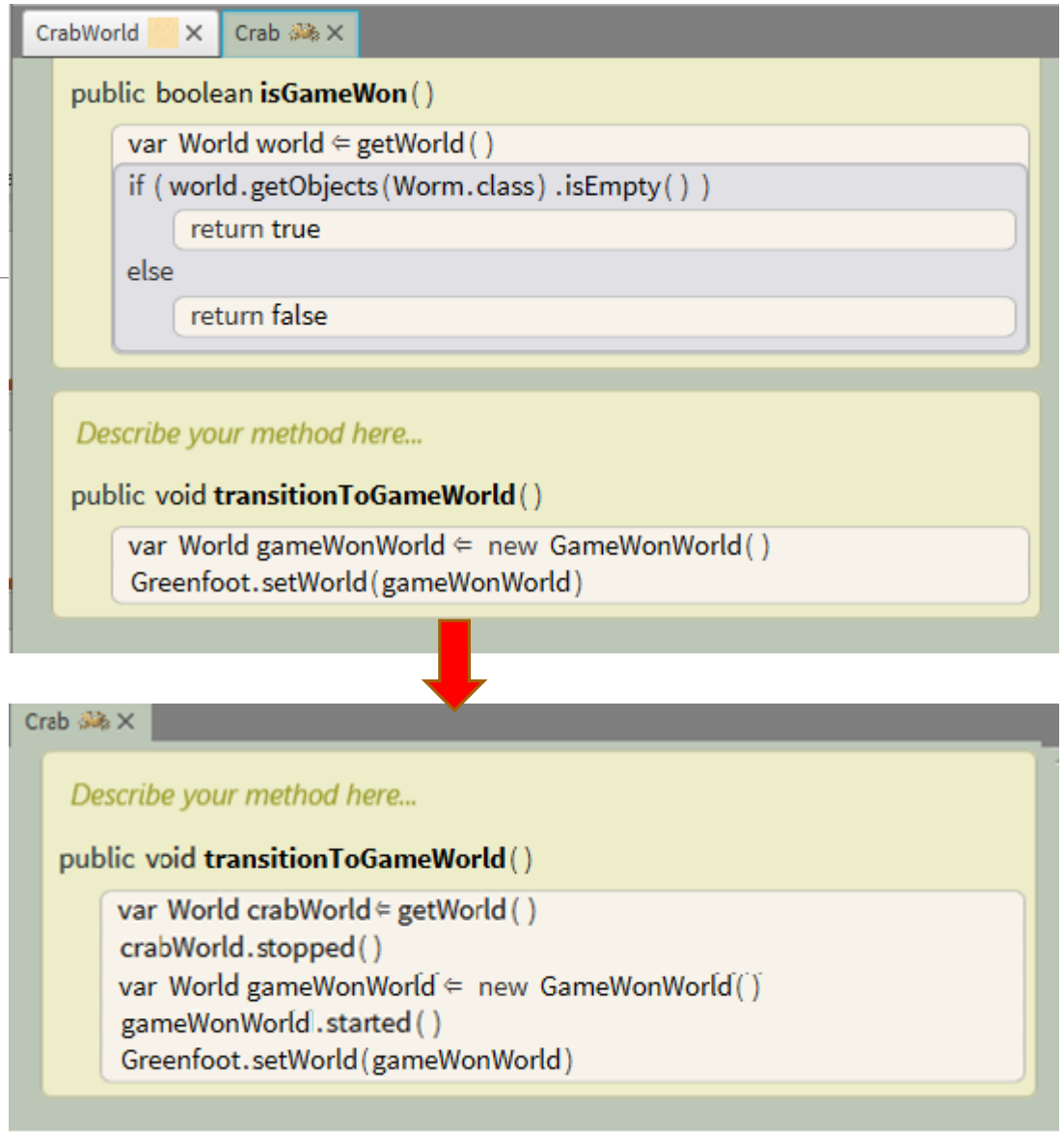


Figure 4a

Step 4
Set up the audio in GameWonWorld

b) Set up the audio in **GameWonWorld** with a different track (winning.wav):

Complete the code in **Figure 4b** to play the sound "winning.wav"

class **GameWonWorld** extends World ▸

Fields

Constructors

Constructor for objects of class GameWonWorld.

```
public GameWonWorld()
    super(560, 560, 1)
    showTextWithBigBlackFont( "You Win" , 180, 200)
```

Methods

Describe your method here...

```
public void showTextWithBigBlackFont(String message, int x, int y)
    var GreenfootImage bg ⇐ getBackground()
        Font font ⇐ new Font(50)
    bg.setFont(font)
    bg.setColor(Color.BLACK)
    bg.drawString(message, x, y)
```

Figure 4b

Step 5: Finish coding the Lobster

After setting up the audio playback in ***GameWonWorld*** with a track (**winning.wav**), set up the audio playback in **GameOverWorld** with a different track (**loosing.wav**):

a) Make transition between worlds (from **CrabWorld** to **GameWonWorld**) with invoking the started() and stopped() method by code

b) Set up the audio in **GameOverWorld** with a different track (loosing.wav)

CrabWorld | Crab | Lobster

```
public void eatWorm()
    var Actor worm ⇐ getOneIntersectingObject(Worm.class)
    if ( worm != null )
        var World world ⇐ getWorld()
        world.removeObject(worm)

Describe your method here...
public boolean isGameLost()
    var World world ⇐ getWorld()
    if ( world.getObjects(Crab.class) .isEmpty() )
        return true
    else
        return false

Describe your method here...
public void transitionToGameOver()
    var World gameOverWorld ⇐ new GameOverWorld()
    Greenfoot.setWorld(gameOverWorld)
```

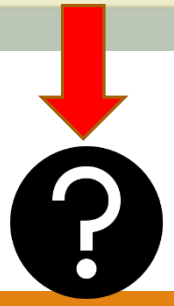


Figure 5

Step 6: Finding music online

Look for Royalty Free music

Some game music have loop markers, they have a lead-in intro and loop at a specific marker without re-playing the intro

More on this later…

https://opengameart.org is a great resource for game music

➢ TO DO

Replace the 3 sounds (**gameplay.wav**, **winning.wav and loosing.wav**) with 3 other sounds of your choice that you search from the aforementioned web site: **https://opengameart.org**

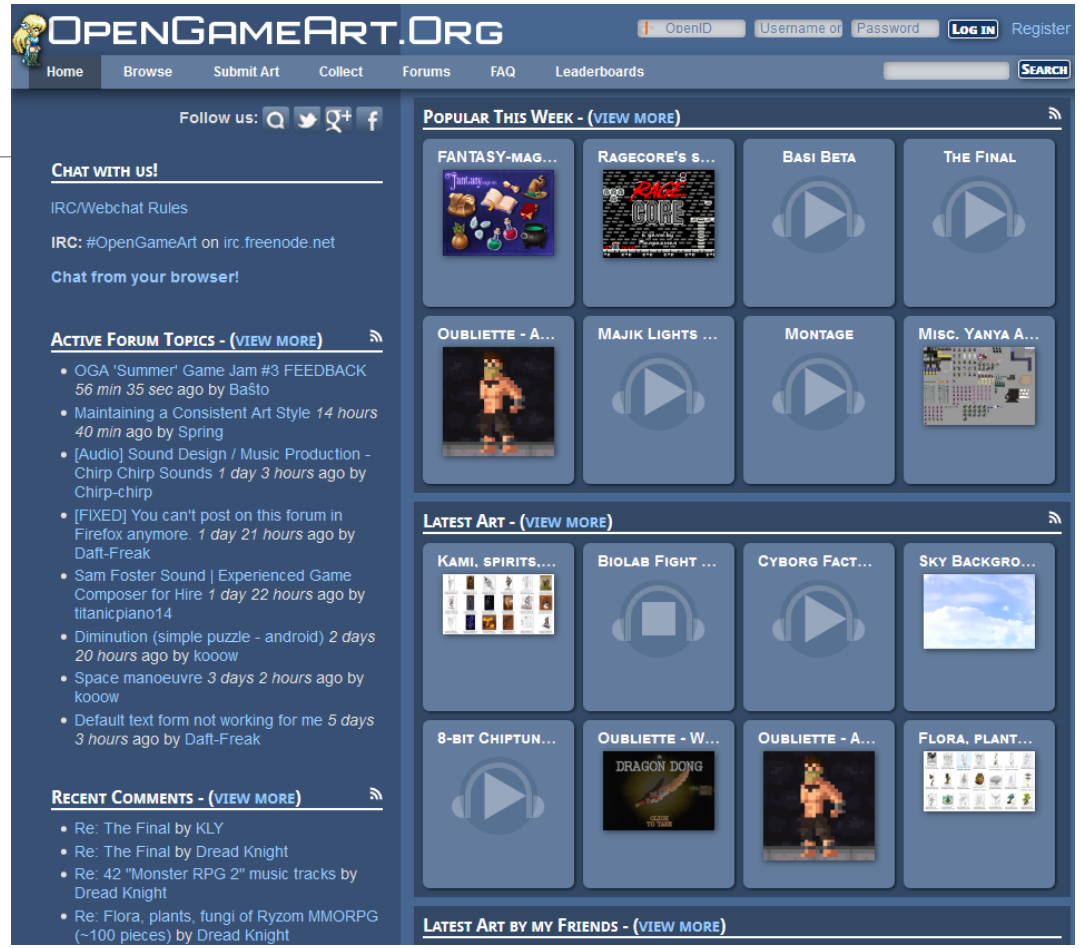


Figure 6

Questions

?