ALGORITHMS AND PROGRAMMING - PRACTICAL EXAM

Deadline: you have **30 minutes** to solve the exercises. After the deadline you can **NOT** upload your solution and you will be marked as absent from the exam.

Exercise: Spotify needs an application to manage the listeners for each song available. There are 2 genres: pop and rock. Each song is characterized by nameOfArtist and numberOfListeners. The application should allow to:

- Update a song **(1p)**. If the number of listeners is not a positive number then the song should not be updated and a message should be displayed **(1p - only if exceptions are used).**
- Identify the most listened song in the pop category as well as the rock category **(2p - only if one single algorithm is used).**
- Print the nameOfArtist for the song with the minimum numberOfListeners for each category. For this requirement, please provide a test function using at least three assertions **(2p).**

Other Requirements:

- Calling the functions implemented in the **main** module. The list should be printed after each operation. (**1p**)
- The initial list should contain at least 5 elements (loaded at the beginning of the application). (**1p**) Example for the content of the repository:

Pop: Song1, 30 // Song2, 200 // Song3, 45 // Song4, 43 // Song5, 61

Rock: Song2, 201 // Song2, 305 // Song3, 20 // Song4, 32 // Song5, 67

- **Style:** clarity of the code (**1p**)
- Use **layered** architecture! Otherwise you will receive maximum the **50% of the final grade**.

Notes:

- You do **NOT** have to have *ui* and *application* modules. You should have a structure like: *practical_exam*

```
├── domain
│  └── song.py
├── infrastructure
│  └── song_repository.py └── main.py
```

TOTAL: 9 points + 1 for appearance