1 General Information about the Project

As part of your course requirement, you are to complete the project described below, which must be carried out individually. Submission of the project for evaluation must be done via email to the address: `francesco.alfieri5@studio.unibo.it`.

The deadline for submission is **23:59:59 hours on 04 January, 2024**.The email must have the subject field as *CSNS Project 2024* and must be sent from your University address (`name.surname@studio.unibo.it`).
You will receive a confirmation message within a few days of your submission. The email should contain an archive (in .zip or .tar.gz format) containing the following:

1. The source code that was developed (either in NetLogo, MatLab or PeerSim);
2. A short paper (up to 12 pages), in PDF format, describing the model that was implemented, the experiments that were carried out using it, and a discussion explaining the results that were obtained.

Your full name, email address and student ID number (matricola) must be included in all of the source files, in the paper, and in the submission email that you send. The source code should be well documented and formatted, following good programming practices. The paper must be written in English, and should be structured like a technical paper, thus containing a title, abstract and bibliography. It is strongly suggested that you follow the Springer format for *Lecture Notes in Computer Science* (LNCS). Templates are available for both Word[1] and LaTeX[2]. You can use any text processing system you prefer (even though LaTeX is suggested) to write the paper as long as you submit the result as a PDF file.

The project must be done *individually*: no sharing of papers or source code is permitted. You are, of course, encouraged to discuss issues and solutions with fellow students or with the instructors.

2 Grading Policy

For your project to be satisfactory, it must satisfy the following requirements:

- The project must implement the specifications that follow. You are allowed (and encouraged) to apply modifications and extensions to the project, but they must be proposed to the instructors beforehand and approved by them.
- The model's implementation, and all of the related simulations and experiments, must be carried out using either the NetLogo, MatLab or PeerSim software systems. If PeerSim is chosen, the cycle-driven simulation engine should be used, and the simulator must be configurable the standard PeerSim configuration file.
- Your paper must thoroughly describe the model that was implemented and justify all significant design decisions and extensions that were applied to it. You should also discuss the expected behavior of the model, by making previsions. Most importantly, you have to explain the experiments that you performed in terms of methodology and the results that you obtained. Significant implementation details can be inserted, if important in the context of the model, but should otherwise be kept as comments in the code itself.

[1] Link to .doc template
[2] Link to .tex template

You are encouraged to focus on a simple model and to apply extensions to it only if you completely understand the behavior of the base model. This can be achieved by working in modular fashion, thus incrementally (and carefully) adding new features, enriching your model. Ending up with a complex, unpredictable and difficult to understand model is very easy. On the contrary, you should prove through your experiments that you fully understand the behavior of your model and that you can interpret the results you obtained, and are able to relate them with real-world phenomena.

3 Introduction to the Project

The purpose of this year's project is to study, implement and analyze the evolution of an epidemic throughout a network with discrete timesteps. In the CSNS course, you have studied contagion and the evolution of epidemics by means of differential equations, for example with SIR, SEIR models and variants. In this project, your goal is to build a simulation of an epidemic with discrete timesteps by means of adjacency matrices.

3.1 Adjacency matrices and network theory

Definition 1. *Suppose $G = (V, E)$ is a network where $V = \{1, 2, \dots, n\}$. For $1 \leq i, j \leq n$ we define*

$$a_{ij} = \begin{cases} 1, & (i,j) \in E \\ 0, & (i,j) \notin E. \end{cases}$$

Then the square matrix $A = (a_{ij})$ is called the adjacency matrix *of G.*

Adjacency matrices are a powerful tool to represent network topologies and processes that take place on a given graph/network. For example, the out-degree of node i is the sum of values in the i-th row of A. Moreover several measures of nodes' centralities can be computed starting from A (the most important ones can be obtained by means of spectral properties of matrices). More useful informations about adjacency matrices can be found in [1].