

# Certamen I

## Lenguajes de Programación II – ICI425/INC415

Profesor: [Alonso Inostrosa Psijas](#)

Fecha: Miércoles 13 de Septiembre de 2023

### Consideraciones:

- Programar usando Flex+Bison o ANTLR.
- Defina las ER, y GLC que sean necesarias.
- Haga uso de tabla de símbolos en caso de ser necesario.
- Grupos de máximo 3 integrantes.
- El profesor puede estimar conveniente reunirse con los integrantes de cada equipo para realizar preguntas sobre el desarrollo del trabajo, de modo que todos los integrantes deben conocer todos los detalles del código, esto puede influir en el factor de participación (FP\_Cert1).
- Fecha de Entrega: 11 de Octubre de 2023.

### Autómatas Celulares:

La teoría de autómatas celulares se inicia con John Von Neumann (finales década 1940), a pesar de que fueron concebidos originalmente por Konrad Zuse y Stanislaw Ulam. La idea de Von Neumann era desarrollar una máquina que fuera capaz de autorreplicarse.

Un autómata celular (AC) (1; 2) es un modelo matemático-computacional para sistemas dinámicos que evolucionan (cambia su estado) en pasos discretos. Permite modelar fácilmente sistemas naturales que pueden ser descritos como entidades que interactúan localmente entre sí.

Consisten en una grilla (normalmente en 2D), donde cada celda corresponde a una célula, la que posee un estado (habitualmente un valor en  $\mathbb{Z}$ ). En modelos AC clásicos, las células se organizan uniformemente en una grilla n-dimensional (ver Figura 1).

(1,1)	(1,2)	(1,3)	(1,4)
(2,1)	(2,2)	(2,3)	(2,4)
(3,1)	(3,2)	(3,3)	(3,4)
(4,1)	(4,2)	(4,3)	(4,4)

Figura 1 - Ejemplo de CA clásico.

Cada célula, además, posee una vecindad compuesta por un conjunto finito de células (ver Figura 2). En cada paso, se aplica un conjunto de reglas (función de transición) a las celdas para determinar su nuevo estado (valor). Un ejemplo de la evolución de un CA en  $n$  pasos se observa en la Figura 3.

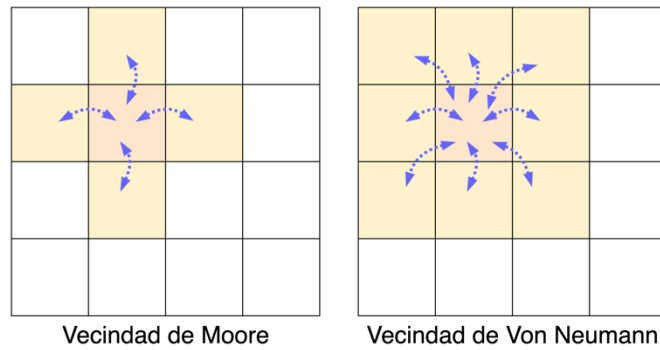


Figura 2 - Tipos de vecindades en AC.

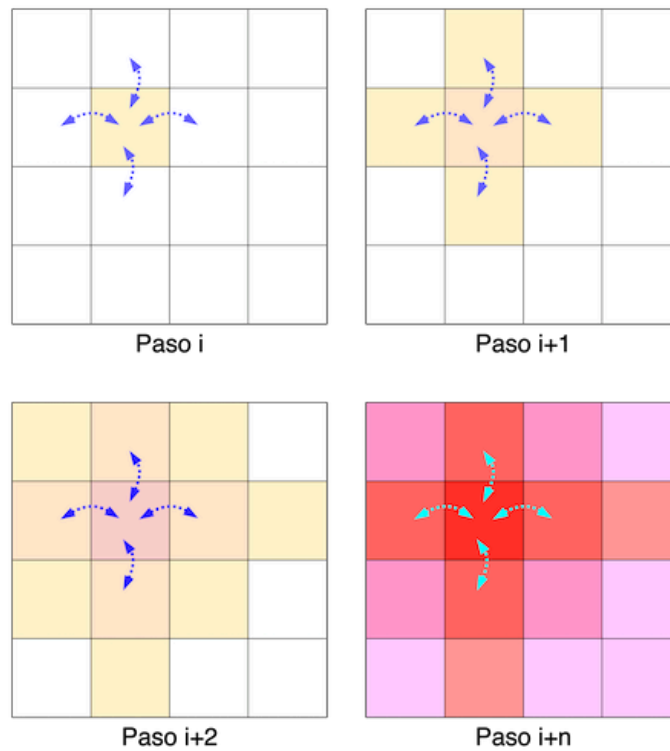


Figura 3 - Evolución de un autómata celular.

Formalmente, un AC se define mediante la siguiente cuádrupla:

$$CA = \langle S^n, \Theta, N, \tau \rangle$$

Donde:

$S^n$  es el espacio n-dimensional.

$\Theta$  es el conjunto de estados. En un paso de tiempo  $t$ , el estado de cualquier celda corresponde a uno de los estados definidos.

$N$  es el conjunto de vecindades, y contiene los vecinos de cada celda.

$\tau$  es la función local de transición (se compone de todas las reglas) y permite cambiar los estados de cada celda.

En AC la evolución del sistema ocurre mediante la ejecución paralela de la función local de transición de cada celda en cada paso de tiempo. Estas funciones, que sólo dependen de los resultados de una ejecución local en cada celda, generan un comportamiento emergente global que puede ser estudiado en modelos de gran escala (3).

## Modelos de Contagio Compartimentales:

En 1927, Kermack y McKendrick (4) introdujeron un modelo para estudiar la dinámica de propagación de la Peste Negra en Londres. Pese a ser un modelo epidemiológico simple, resultó ser altamente efectivo, razón por la que es uno de los más utilizados (5). Su sencillez radica en dividir a la población en compartimentos relacionados con los estadios de la enfermedad por los que deben pasar los individuos de una población. En consecuencia, los individuos de la población pueden ser Susceptibles (S), Infectados (I) o Recuperados (R). En este modelo SIR, se asigna al compartimento S a una persona que nunca se ha infectado. Los individuos que adquirieron la enfermedad y pueden infectar a otros se asignan al grupo I. Aquellos que se han recuperado y ahora son inmunes (por algún tiempo) a la enfermedad se asignan al compartimento R. Los modelos contienen un conjunto de parámetros que afectan su dinámica. Diferentes combinaciones de parámetros pueden conducir a escenarios que son catastróficos y otros en los que la enfermedad remite naturalmente (6).

Los modelos SIR se han ampliado y adaptado, añadiendo más ecuaciones que representan nuevos compartimentos o características. Por ejemplo, los modelos SEIR consideran un compartimento para representar a los individuos Expuestos (E) a enfermedades, y corresponden a aquellos que, al contraer la enfermedad, pasan por un período de incubación antes de estar infectados (7). Su dinámica se describe mediante el diagrama de estado de la Figura 4. Las aristas corresponden a las funciones asociadas al cambio de una etapa de la enfermedad a otra:  $\Lambda$  es la tasa de reclutamiento (es decir, nacimientos),  $\beta$  es la tasa de infección,  $\alpha_1$  es la tasa de morbilidad<sup>1</sup>,  $\alpha_2$  la tasa de recuperación,  $\rho$  es la tasa de pérdida de inmunidad,  $\mu$  la tasa de mortalidad por causas distintas a la enfermedad, y  $\delta$  es la tasa de mortalidad inducida por la enfermedad.

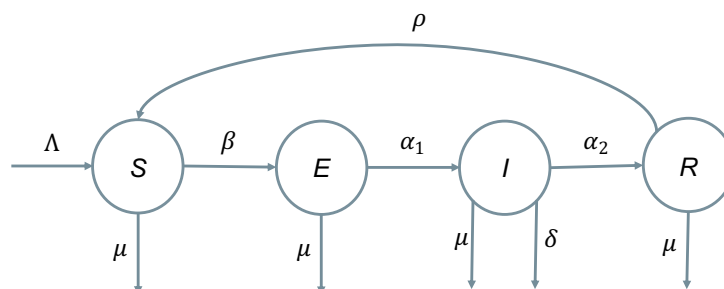


Figura 4 - Diagrama de transición de estados del modelo SEIR (8).

Un sistema de ecuaciones diferenciales ordinarias puede derivarse diagrama de la Figura 4, lo que deriva en:

$$\frac{dS}{dt} = \Lambda + \rho R(t) - \beta S(t) - \mu S(t); \frac{dE}{dt} = \beta S(t) - \alpha_1 E(t) - \mu E(t); \frac{dI}{dt} = \alpha_1 E(t) - \alpha_2 I(t) - \mu I(t) - \delta I(t); \frac{dR}{dt} = \alpha_2 I(t) - \rho R(t) - \mu R(t).$$

En el contexto de las epidemias, varios modelos AC han utilizado enfoques basados en SIR para modelar la propagación de enfermedades contagiosas (9), donde cada celda del AC puede representar una zona geográfica con población en diferentes estadios dentro del modelo de contagio. En otros casos, cada celda representa simplemente a un individuo, sin embargo, este tipo de enfoques posee la limitante de que no resulta eficiente en la simulación de modelos con un tamaño de población realista.

<sup>1</sup> Cantidad de personas que enferman en un lugar y un período de tiempo determinados en relación con el total de la población.

## Descripción del Trabajo:

1. Implemente un lenguaje de programación que permita la creación y manipulación de AC multicapa (ver Figura 5) para la simulación modelos de contagio compartimentales. Un AC multicapa puede definirse a partir de la conexión de diferentes AC bidimensionales. En el AC de la Figura 5, los diferentes AC bidimensionales corresponden (y componen) un AC multicapa de dimensión  $n$ . De este modo, la vecindad de una célula puede estar conformada por celdas de su misma capa y de las capas directamente adyacentes. Para representar el contagio, se debe determinar la vecindad de cada célula, y la forma en que la célula podría infectarse de acuerdo al estados de su vecindad. En la Figura 6 se observa una vecindad de Moore para el AC de la Figura 5.

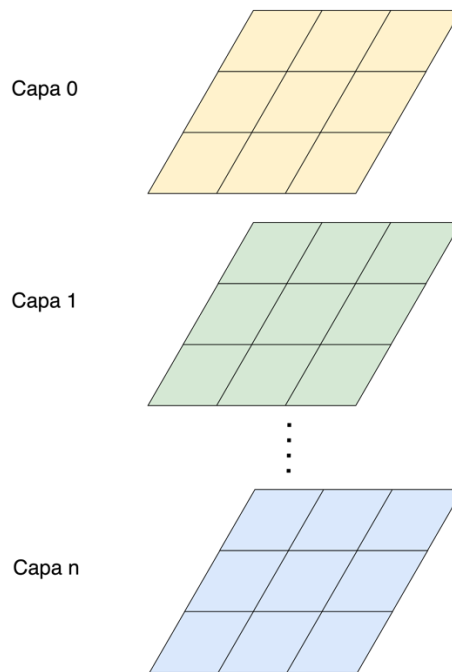


Figura 5 - AC multicapa.

2. Con el lenguaje desarrollado (y su respectivo intérprete) deberá permitir la simulación del contagio de una enfermedad contagiosa en un espacio geográfico (definido por el AC multicapa). Cada celda del AC representará un individuo, donde cada individuo de la población se encontrará en alguno de los estados de la enfermedad de acuerdo con un de modelo SIR de su elección (ej: SEIR, SIRD, SEIRD, etc). **No se permite el modelo SIR.**
3. Con el lenguaje desarrollado (y su respectivo intérprete) deberá permitir la simulación del contagio de una enfermedad contagiosa en un espacio geográfico (definido por el AC multicapa). Cada celda del AC representará un área geográfica con una población determinada, donde cada individuo de la población de cada celda se encontrará en alguno de los estados de la enfermedad de acuerdo con un de modelo SIR de su elección (ej: SEIR, SIRD, SEIRD, etc). **No se permite el modelo SIR.**

4. Utilizando el lenguaje creado, extienda el punto 3 para construir y explicar un AC de ejemplo que permita simular dos escenarios diferentes:
  - I. Uno donde exista tránsito entre individuos de celdas vecinas (una proporción de los individuos puede transitar a otra celda).
  - II. Un escenario donde hay confinamiento de la población, restringiendo su movimiento sólo a su zona geográfica (celda).
5. Opcional: Defina mecanismos que permitan visualizar la simulación del contagio entre los habitantes dentro de una misma célula y entre células vecinas. Si le resulta más conveniente, la visualización puede realizarla de forma posterior a la simulación. Es decir, mediante la simulación puede generar una traza y luego, utilizando algún software de su preferencia, puede generar la visualización de la traza

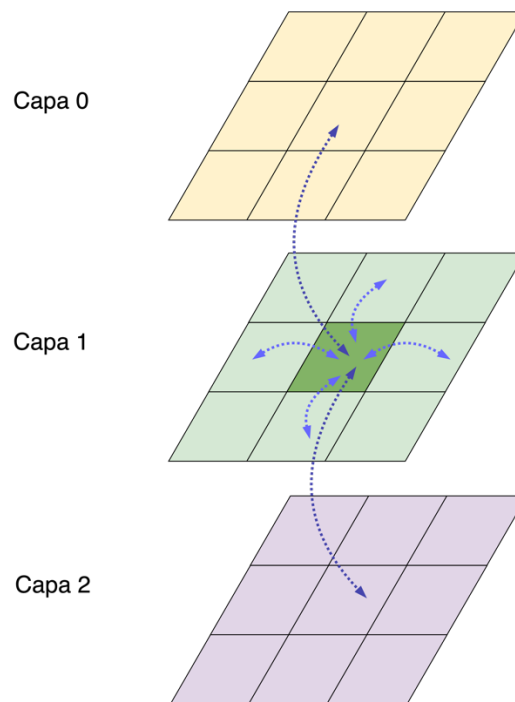


Figura 6 - Vecindad de Moore en un AC multicapa.

## Entrega:

El entregable se compone de:

- Código implementado con su respectivo respaldo en un repositorio Github.
- Presentación en clases de 15 minutos sobre su solución y ejemplo.

El entregable debe ser subido en un único archivo .zip al Aula Virtual, en el recurso respectivo, con el siguiente formato:

Certamen-1-ApellidoIntegrante1-ApellidoIntegrante2.zip

Ejemplo: Certamen1-Alvarez-Perez-Soto.zip

Todos los integrantes del grupo deben comprender y poder explicar en su totalidad el código, por lo que deben repartirse trabajo y comprender tanto la teoría como la práctica. **La copia entre equipos de trabajo está estrictamente prohibida y será amonestada con nota 1.0 para cada grupo que la realice.**

## Sistema de Evaluación:

Cada estudiante debe responder una encuesta que se le hará llegar, indicando el factor de participación (FP\_Cert1) para cada compañero/a de grupo. La nota del Certamen 1 estará dada por una nota grupal (NotaGrupal1) resultante de los programas implementados, y un factor de participación (FP\_Cert1) entregado por sus pares de forma anónima:

$$\text{Nota Certamen 1} = \text{NotaGrupal1} * \text{FP\_Cert1}$$

## Nota Grupal:

Criterio	Subtotal Puntos
<b>Calidad de Código:</b> <b>El código DEBE compilar, de lo contrario no será revisado.</b> Los códigos implementados se ejecutan sin errores. Los códigos son claros, está bien documentados y se ajustan a las condiciones de entrega. Los datos de entrada y salida están comentados en el archivo readme.txt.	5 Pts.
<b>Presentación:</b> <ol style="list-style-type: none"> <li>1. Presentación (considera estructura, ortografía y redacción): 5Pts.</li> <li>2. Introducción y Problemática: 5 Pts.</li> <li>3. Diseño de la solución propuesta: 30pts.               <ol style="list-style-type: none"> <li>a. Explicación de la sintaxis, incluyendo ejemplos: 10 Pts.</li> <li>b. Explicación de funciones para creación de AC, proveer ejemplos: 10 Pts.</li> <li>c. Análisis de resultados de los escenarios simulados: 10.</li> </ol> </li> <li>4. Conclusiones: 5Pts.</li> </ol>	45 Pts.
<b>Correctitud y completitud</b> <ol style="list-style-type: none"> <li>a. Correctitud y completitud de la pregunta 1: 10 Pts.</li> <li>b. Correctitud y completitud de la pregunta 2: 15 Pts.</li> <li>c. Correctitud y completitud de la pregunta 3: 15 Pts.</li> <li>d. Correctitud y completitud de la pregunta 4: 10 Pts.</li> <li>e. <i>(Opcional) Correctitud y completitud de la pregunta 5: Extra 15 Pts.</i></li> </ol>	50 Pts.
<b>Total</b>	<b>100 Pts.</b>

## Factor de Participación (FP\_Cert1):

La nota FP\_Cert1 para un/a estudiante corresponde al promedio de los factores ingresados por sus compañeros/as. Eventualmente, también pueden incidir las respuestas a preguntas realizadas por el profesor, en caso de que el grupo se entreviste con éste.

Nivel de Participación	Factor
<b>Excelente:</b> Mi compañero/a fue indispensable para realizar del trabajo. Sin él/ella no se podría haber realizado (solo puede evaluar a máximo 1 integrante de esta manera).	1.1

<b>Bueno:</b> Mi compañero/a se desempeñó de manera satisfactoria y responsable en el equipo.	<b>1.0</b>
<b>Insuficiente:</b> Mi compañero/a no participó lo suficientemente en el trabajo. Su desempeño no fue el ideal esperado.	<b>0.8</b>
<b>Deficiente:</b> Mi compañero/a no fue un aporte al equipo. El equipo habría funcionado mejor sin él/ella.	<b>0.6</b>

## Referencias

1. *Cellular automata as models of complexity*. **Wolfram, Stephen**. 5985, s.l. : Nature Publishing Group, 1984, Nature, Vol. 311.
2. *Asymmetric Cell-DEVS models with the Cadmium simulator*. **Cárdenas, Román and Wainer, Gabriel**. s.l. : Elsevier, 2022, Simulation Modelling Practice and Theory, Vol. 121.
3. **Wolfram, Stephen**. *A new kind of science*. Champaign, IL : Wolfram media, 2002.
4. *A contribution to the mathematical theory of epidemics*. **Kermack, William Ogilvy and McKendrick, Anderson G**. s.l. : The Royal Society of London, 1927. Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character. pp. 700-721.
5. *Epidemics and rumours: A survey*. **Dietz, Klaus**. 4, s.l. : Wiley Online Library, 1967, Journal of the Royal Statistical Society: Series A (General), Vol. 130, pp. 505-528.
6. *Function estimation and regularization in the SIRD model applied to the COVID-19 pandemics*. **Pacheco, C. C. and de Lacerda, C. R**. s.l. : Taylor & Francis, 2021, Inverse Problems in Science and Engineering, pp. 1-16.
7. *Seasonality and period-doubling bifurcations in an epidemic model*. **Aron, Joan L and Schwartz, Ira B**. 1984, Journal of theoretical biology, pp. 665-679.
8. *A Modeling and Simulation Platform for Space-Based Compartmental Modeling of Pandemic Spread*. **Cárdenas, Román, Inostrosa-Psijas, Alonso and Wainer, Gabriel**. Fairfax, VA, USA : s.n., 2021. 2021 Annual Modeling and Simulation Conference (ANNSIM).
9. *A simple model of recurrent epidemics*. **Johansen, Anders**. 1, s.l. : Elsevier, 1996, Journal of theoretical biology, Vol. 178, pp. 45-51.
10. **Hopcroft, John E., Motwani, Rajeev and Ullman, Jeffrey D**. *Introduction to automata theory, languages, and computation*. New York, NY, USA : ACM, 2001.
11. *Automata for modeling real-time systems*. **Alur, Rajee and Dill, David**. s.l. : Springer, 1990. International colloquium on automata, languages, and programming. pp. 322-335.
12. *Timed automata: Semantics, algorithms and tools*. **Bengtsson, Johan and Yi, Wang**. s.l. : Springer, 2003. Advanced Course on Petri Nets.
13. *Modeling epidemics using cellular automata*. **Hoya White, S., Martín del Rey, A. and Rodríguez Sánchez, G**. 2007, Applied mathematics and computation, pp. 193-202.
14. *Simulation of the spread of infectious diseases in a geographical environment*. **Zhong, ShaoBo, Huang, QuanYi and Song, DunJiang**. 4, s.l. : Springer, 2009, Science in China Series D: Earth Sciences, Vol. 52, pp. 550-561.
15. *Spread of Infectious Disease Modeling and Analysis of Different Factors on Spread of Infectious Disease Based on Cellular Automata*. **Bin, Sheng, Sun, Gengxin and Chen, Chih-Cheng**. 23, s.l. : Multidisciplinary Digital Publishing Institute, 2019, International journal of environmental research and public health, Vol. 16, p. 4683.

