

---

# Informa2 S.A.S.

septiembre de 2023

## Presentación

Es muy común por estos días que los negocios con sede física tengan fachadas llamativas para atraer más clientes a sus negocios. Para esto, y entendiendo que hay que generar estrategias que permitan al cliente buscar su negocio de una manera más fácil y además generar recordación, se usan matrices de LEDs para imprimir mensajes en la fachada que muestre ya sea que el local se encuentra abierto o que se ofrece algún producto. Un ejemplo de esto se presenta a continuación.



## Objetivos

1. Desarrollar la capacidad de solución de problemas en los estudiantes enfrentándolos a problemáticas de la vida cotidiana.
2. Identificar si el estudiante adquirió las destrezas y conocimientos fundamentales de la programación con C++, en donde resaltamos estructuras de programación (iterativas, secuenciales y decisión), tipos de datos, apuntadores, arreglos y funciones.
3. Evaluar la capacidad del estudiante para trabajar con Arduino e integrar la programación en C++, además de usar de manera adecuada las funciones de la plataforma.

---

## Instrucciones

El objetivo principal de esta actividad es poner a prueba sus destrezas en análisis de problemas y manejo del lenguaje C++. Si usted ha llevado un proceso disciplinado de aprendizaje a lo largo del semestre, esta es una oportunidad de demostrarlo, podrá plantear una solución viable y su resultado será satisfactorio. En caso contrario, podrá identificar sus debilidades y deberá tomar medidas con el fin de poder ser apto para enfrentar situaciones como las que se plantean en la sección Presentación.

Trate en lo posible de valorar muy bien la verdadera complejidad del problema planteado, no se rinda antes de intentarlo o de plantear los posibles escenarios de solución. Se dará cuenta que si bien, al principio le puede parecer largo y difícil, realmente ya ha tenido la oportunidad de enfrentarse a problemas similares. Si se toma el tiempo adecuado para analizar, el proceso de codificación no le tomará mucho tiempo.

Esperamos que disfrute del desafío propuesto. Le recomendamos que lea primero todo el documento antes de comenzar y asegúrese de entender muy bien las instrucciones antes de desarrollar esta actividad evaluativa.

La evaluación de esta prueba junto con un quiz, representan el 20% de la nota final del curso. Fue revisado por los profesores Aníbal Guerra y Augusto Salazar.

## Requisitos mínimos

A continuación, se describen los requisitos que se deben cumplir. El incumplimiento de cualquiera de ellos implica que su nota sea cero.

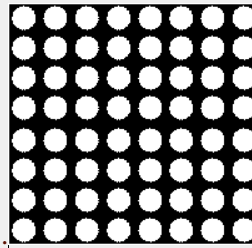
1. Genere un informe en donde se detalle el desarrollo del proyecto, explique entre otras cosas:
  - a. Análisis del problema y consideraciones para la alternativa de solución propuesta.
  - b. Esquema donde describa las tareas que usted definió en el desarrollo de los algoritmos.
  - c. Algoritmos implementados.
  - d. Problemas de desarrollo que afrontó.
  - e. Evolución de la solución y consideraciones a tener en cuenta en la implementación.
2. La solución debe ser implementada en lenguaje C++.
3. La implementación debe incluir el uso de punteros, arreglos y memoria dinámica.
4. El programa debe ser una aplicación desarrollada en la plataforma TINKERCAD.
5. Se debe crear un repositorio público en el cual se van a poder cargar todos los archivos relacionados a la solución planteada por usted (informe, código fuente, enlace al proyecto de TINKERCAD y otros anexos).
6. Una vez cumplida la fecha de entrega no se podrá hacer modificación alguna al proyecto de TINKERCAD.
7. Una vez se cumpla el plazo de entrega, no se deberá hacer modificación alguna al repositorio, en caso contrario su nota es cero.
8. Se deben hacer *commits* de forma regular (al menos dos al día) de tal forma que se evidencie la evolución de la propuesta de solución y su implementación.
9. Se debe adjuntar un enlace de *youtube* a un video que debe incluir lo siguiente:
  - Presentación de la solución planteada. Análisis realizado y explicación de la arquitectura del sistema (3 minutos máximo).
  - Demostración de funcionamiento del sistema. Explicar cómo funciona en tiempo de simulación (3 minutos máximo).
  - Explicación del código fuente. Tenga en cuenta que debe justificar la elección de las variables y estructuras de control usados. Por qué eligió uno u otro tipo de variable o estructura de control en cada caso particular y que ventaja ofrecen estos en comparación de otras que también podrían haber sido usados (5 minutos máximo).

- La duración total del video no debe exceder 11 minutos ni ser inferior a 5 minutos.
  - Asegúrese que el video tenga buen sonido y que se puede visualizar con resolución suficiente para apreciar bien los componentes presentados.
10. El plazo de entrega se divide en dos momentos:
    - a. El día 18 de septiembre para adjuntar la evidencia del proceso de análisis y diseño de la solución.
    - b. El día 25 de septiembre para adjuntar la evidencia del proceso de implementación
  11. Se deben adjuntar **dos enlaces**: uno al repositorio y otro al video, nada más.

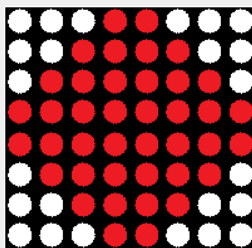
## Especificaciones

En orden con la presentación y los objetivos necesarios para la solución de la necesidad planteada en la sesión de explicación, a continuación, se presentan los requerimientos:

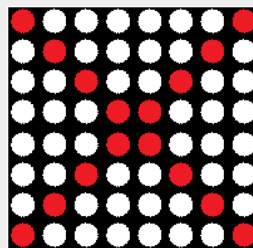
1. **[0.5]** Realice las conexiones necesarias para tener en estado operativo el sistema compuesto por 64 LEDs conectados siguiendo la siguiente estructura:



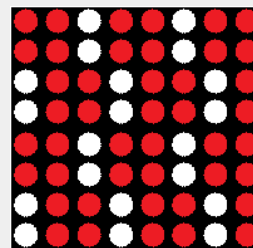
2. **[0.5]** Realice una función llamada **verificación** que le permita al usuario comprobar que todos los LEDs de la matriz de 8x8 están funcionando correctamente. Para esto, encienda y apague los 64 LEDs cada cierto tiempo.
3. **[1.0]** Desarrolle una función llamada **imagen**, que le permita al usuario mostrar un patrón en la matriz de LEDs. El usuario debe poder ingresar el patrón por el monitor serial.
4. **[1.0]** Escriba una función que genere cada uno de los patrones mostrados a continuación.



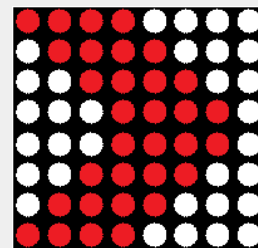
Patrón 1



Patrón 2



Patrón 3



Patrón 4

Fíjese que los patrones se pueden generar a partir de secuencias. La solución debe ser de esta manera, no de forma trivial predefiniendo que LEDs van encendidos en cada caso.

5. **[1.0]** Desarrolle una función llamada **publik**, que le permita al usuario seleccionar (con comandos enviados por el puerto serial):
  - a. Verificar funcionamiento de LEDs. Indicando cuanto tiempo entre encendido y apagado y cuantas secuencias.
  - b. Mostrar imagen de prueba. Ingresada por el puerto serial y prendiendo y apagando según el tiempo indicado por el usuario
  - c. Mostrar de forma alternada los patrones 1 al 4. Indicar el tiempo de retardo entre visualizaciones. Los patrones se deben generar a partir de secuencias como en el punto 4.

---

## Restricciones y recomendaciones

Su solución deberá usar máximo 7 pines digitales del Arduino. Por cada pin adicional que utilice se le penalizará con -0.5.

No se pueden usar *NeoPixel* en la implementación

Investigue la propiedades y funcionamiento del integrado 74HC595 y considere usarlo en su implementación.