|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UASD/ 2021-20/ INF-512 e INF513/ Prof. Julio Castro | | 202111 | | Lab04 |
| NOMBRE |  | MATRICULA  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  |  |  |  |  |  |  |  |  | | | Para uso del profesor |
| APELLIDO |  |
| SECCIÓN |  |

**TEMA:** Constructores y Destructores

**REQUERIMIENTO**

Diagram, box and whisker chart

Description automatically generated

Usted ha sido contratado para escribir el software de un elevador, cuyas funciones principales consisten en subir y bajar pisos seleccionados por una o varias personas. Al llegar al piso requerido, el elevador se detiene y la puerta se abre. Para continuar al siguiente piso requerido, la puerta se cierra y el elevador continúa. Un elevador no puede subir o bajar si la puerta está abierta. El elevador debe indicar por cual piso va cuando se encuentra en operación.

Todo elevador tiene un piso inicial y un piso tope. Los elevadores no pueden ir más allá de su piso tope. Los elevadores de este ejercicio son de la marca **ELEVATORWELL**.

**ELEVATORWELL** incluye funcionalidades de encendido y apagado automático. Para este ejercicio, se apagará automáticamente en las noches y de igual forma encenderá en las mañanas. Cada vez que ELEVATORWELL se enciende, se dirige al piso inicial, abre la puerta y despliega el mensaje “*Bienvenido a ELEVATORWELL. Póngase cómodo y seleccione un piso hacia donde quiere ir*”.

**Usted debe:**

1. (30 %) - Identificar la clase principal que interviene en este problema y escribir su notación UML.
2. (40 %) - Escribir el código fuente C# que implementa la clase identificada en el punto anterior.
3. (30%) - Escribir el código fuente C# de la Tester-class que prueba la clase implementada.

**NOTA**: 2 y 3 se deben escribir en un mismo archivo llamado ELEVATORWELL.cs

**Consejo:** siéntete libre de la utilizar creatividad para resolver este laboratorio. Recuerda mantener la sencillez del concepto para que cumplas con el propósito de mostrar tu conocimiento sobre los constructores y destructores. Recuerda también utilizar el concepto de encapsulación para controlar los valores de los atributos de la clase.