## **Integrantes:**

Carlos tafurt

Martin Correa

Carlos Zuleta

## 

## **Definición del problema a resolver:**

La gente de Discreet Guys Inc. necesitan un modelo que simule el funcionamiento de unos ascensores que serán instalados en unos edificios nuevos de la universidad ICESI. El plan es construir diferentes tipos de edificios inteligentes.

Los edificios tendrán varias oficinas en cada piso, la cantidad de oficinas dependerá de los niveles que tenga el edificio, es decir, si el edificio cuenta con 8 niveles tendrá 8 oficinas por piso y el número de identificación de la oficina será ascendente ubicándose en el último piso las oficinas de la 1 a la 8 y en el primer piso las oficinas 56 a la 64.

Para finalizar, la cantidad de personas será determinada para cada nivel, pero puede variar entre pisos.

## **Conceptos investigados:**

1. Stack: Un contenedor de objetos donde se pueden insertar y extraer elementos según el principio LIFO
2. Queue: Son contenedores de objetos donde se pueden insertar y extraer elementos siguiendo el principio FIFO.
3. Generics: Una clase, interfaz o método que funciona con un tipo de parámetro se denomina genérico, como una clase genérica o método genérico. Una ventaja principal del código genérico es que trabajará automáticamente con el tipo de datos pasados a su parámetro de tipo.
4. Hash: Un HashMap básicamente designa claves únicas para los valores correspondientes que se pueden recuperar en cualquier punto dado, asignando así una clave única para cada dato.
5. Colas de prioridad: Una cola especial en la que todos los elementos de la cola se ordenan según el orden natural o utilizando un comparador personalizado se denomina cola de prioridad

## **Requerimientos que debe satisfacer la solución:**

La solución brindada a Discreet Guys Inc. Debe estar en capacidad de:

1. Simular el funcionamiento de estos edificios inteligentes, mostrando en la consola absolutamente todos los movimientos de los ascensores.
2. Los ascensores deben dirigirse a los pisos de acuerdo al orden en el que fueron presionados los botones.
3. El programa debe permitir que cada persona vaya a una oficina determinada y se debe poder operar hacia dónde se dirige cada persona.
4. Cuando las personas estén ingresando al ascensor, se determinará de acuerdo al orden de llegada, la salida será lo inverso, es decir, los que llegan de primeros entran de primeros pero los que entran de últimos son los primeros en salir

## **Propuestas de solución resultantes en la lluvia de ideas:**

1. Hacer uso de stack, queue y colas de prioridad para simular el funcionamiento de un ascensor real, así se cumplirán los requerimientos 2 y 4, de acuerdo al enunciado del problema.
2. Hacer la simulación del ascensor en una interfaz gráfica. Así mismo realizar todos los procedimientos en el main para enviar un solo archivo a la empresa Discreet Guys inc.

**Elegir la mejor solución:**

La mejor solución es la No 1, es la más efectiva y la más completa, pues las funciones a implementar en el programa están en capacidad de satisfacer la necesidad que hay del movimiento del ascensor en el orden que se presionen los botones, de igual forma, queue permite cumplir con la necesidad de realizar una salida y una entrada al ascensor en sentido inversa.