

# Virus Expansion - Análisis

## Quiz 2

### Contexto

Se debe realizar un programa para la OMS que simule la expansión de un virus llamado flojera aguda. Tiene una posibilidad de 90% de contagio y luego de que una persona se infecte, tiene 14 segundos (días) para recuperarse. Las personas serán representadas por círculos pequeños que se moverán por la pantalla, y tendrán tres estados distintos que corresponden a sano, infectado y recuperado, representados por colores distintos: verde, rojo y azul respectivamente. No se contemplan muertes ni re-infecciones. Deben haber indicadores que muestren esta información, que se puedan organizar por color y el número de personas en cada estado. La información inicial de los estados se cargará a través de un archivo .txt.

### Requerimientos

#### Requerimientos funcionales

##### RF1

Descripción	El programa debe poder <b>cargar la información</b> del archivo .txt en una lista.
Entradas	Archivo .txt y arreglo de texto.
Salidas	
Precondición	Que se haya iniciado el programa
Postcondición	Cadena de texto cargada.

##### RF2

Descripción	El programa debe poder <b>crear las personas</b> en sus estados (sano, infectado, recuperado) con la información obtenida del archivo .txt
Entradas	Cadena de texto.
Salidas	
Precondición	Que se haya iniciado el programa y la información del .txt haya sido cargada en la lista.
Postcondición	Personas creadas con sus respectivos estados.

### RF3

<b>Descripción</b>	<i>El programa debe <b>agregar a las personas en una lista</b>.</i>
<b>Entradas</b>	Persona.
<b>Salidas</b>	
<b>Precondición</b>	Que se haya iniciado el programa y se hayan creado las personas en sus respectivos estados.
<b>Postcondición</b>	Persona agregada a la lista de personas.

### RF4

<b>Descripción</b>	<i>Cada persona deberá moverse por la pantalla a través de la ejecución de un <b>hilo</b>.</i>
<b>Entradas</b>	Posiciones de las personas.
<b>Salidas</b>	
<b>Precondición</b>	Que se haya iniciado el programa y la lista de personas no esté vacía.
<b>Postcondición</b>	Persona moviéndose.

### RF5

<b>Descripción</b>	<i>El programa debe <b>crear indicadores</b> de las personas sanas, infectadas, y recuperadas, y deben tener los mismos colores que el estado de persona que representan (verde, rojo, azul).</i>
<b>Entradas</b>	Lista de personas.
<b>Salidas</b>	
<b>Precondición</b>	Que se haya iniciado el programa y la lista de personas no esté vacía.
<b>Postcondición</b>	Indicadores creados y visualizados en la interfaz del programa.

**RF6**

<b>Descripción</b>	<i>El programa debe <b>cambiar el estado</b> de una persona <b>sana a infectada</b> si esta se acerca a una persona infectada, con una probabilidad de 90% a través de un hilo.</i>
<b>Entradas</b>	Posiciones y estados de las personas.
<b>Salidas</b>	Cambio de estado
<b>Precondición</b>	Que se haya iniciado el programa y la lista de personas no esté vacía.
<b>Postcondición</b>	Persona sana, ahora infectada.

**RF7**

<b>Descripción</b>	<i>El programa debe <b>cambiar el estado</b> de una persona <b>infectada a recuperada</b> 14 segundos después de haberse infectado a través de un hilo.</i>
<b>Entradas</b>	Estado de la persona
<b>Salidas</b>	Cambio de estado
<b>Precondición</b>	Que se haya iniciado el programa y la persona haya sido infectada.
<b>Postcondición</b>	Persona infectada, ahora recuperada.

**RF8**

<b>Descripción</b>	<i>El programa debe lanzar una <b>excepción</b> propia que diga <b>"Contagiado"</b> cada que una persona se infecte imprimir el mensaje en consola.</i>
<b>Entradas</b>	Estado de la persona.
<b>Salidas</b>	Excepción
<b>Precondición</b>	Que se haya iniciado el programa y la lista de personas no esté vacía.
<b>Postcondición</b>	Mensaje impreso en consola.

**RF9**

<b>Descripción</b>	<i>El programa debe lanzar una <b>excepción</b> propia que diga “<b>30% de personas infectadas</b>” si el porcentaje de infectados es igual o mayor al 30% de las personas.</i>
<b>Entradas</b>	Lista de personas.
<b>Salidas</b>	Excepción
<b>Precondición</b>	Que se haya iniciado el programa y la lista de personas no esté vacía.
<b>Postcondición</b>	

**RF10**

<b>Descripción</b>	<i>El programa debe agregar los indicadores a una lista de indicadores.</i>
<b>Entradas</b>	Indicadores.
<b>Salidas</b>	
<b>Precondición</b>	Que se haya iniciado el programa y los indicadores hayan sido creados.
<b>Postcondición</b>	Indicadores agregados a la lista de indicadores.

**RF11**

<b>Descripción</b>	<i>El programa debe poder <b>organizar los indicadores naturalmente</b> por número de personas en el respectivo estado (sano, infectado, recuperado).</i>
<b>Entradas</b>	Lista de indicadores.
<b>Salidas</b>	Lista de indicadores organizada por número de personas en cada indicador.
<b>Precondición</b>	Que se haya iniciado el programa y la lista de indicadores no esté vacía.
<b>Postcondición</b>	Lista organizada según criterio en la interfaz.

**RF12**

<b>Descripción</b>	<i>El programa debe poder <b>organizar los indicadores parcialmente</b> por color según el respectivo estado (verde, rojo, azul).</i>
<b>Entradas</b>	Lista de indicadores.
<b>Salidas</b>	Lista de indicadores organizada por color según el respectivo estado.
<b>Precondición</b>	Que se haya iniciado el programa y la lista de indicadores no esté vacía.
<b>Postcondición</b>	Lista organizada según criterio en la interfaz.

**Requerimientos no funcionales**

- El diseño del prototipo será en el lenguaje de programación Java.
- Se realizará con el patrón de diseño MVC (Model, View, Controller).
- Deberá hacer uso de herencia.
- Los elementos del prototipo deberán ser hechos con figuras geométricas.
- Las personas deberán ser representadas por bolitas con 7px de diámetro.