

PARTE 3: Ejercicio de programación JAVA

Duración 60 minutos

Este ejercicio comprende **2** actividades (no se especifica un orden de las mismas):

- a) Desarrollo de la funcionalidad especificada más abajo
- b) Desarrollo de los casos de prueba ("test case", al menos uno) para verificar la corrección de la funcionalidad implementada.

ESCENARIO

En el contexto de una pandemia muy agresiva – COVID19 -, se ha desarrollado una aplicación capaz de registrar los contactos entre las personas, a fin de poder, en el caso de que alguna de ellas se haya infectado, realizar tareas de trazabilidad.

El sistema almacena los contactos registrados entre dos personas cualquiera, entre las que se asume que ha habido contacto suficiente para que el contagio de la enfermedad pueda existir, en caso de que una de ellas se encuentre cursando la enfermedad.

A partir de estos datos, una de las funcionalidades que entonces se quiere desarrollar es que, al indicar que una cierta persona se ha verificado como COVID19 positiva, se analicen las vías de trazabilidad hacia todas las demás personas registradas en la red, y se emitan reportes que indiquen qué personas se encuentran en cada nivel de "distancia de contacto" (distancia de contacto 1, 2, etc.)

PREPARACION

- a) Descarga de la webasignatura el proyecto completo "**Parcial2-2020.zip**" – adjunto en la tarea "**Parcial2 – PARTE 3**" y descomprímelo en una carpeta del disco duro local.
- b) Abre el Proyecto en NETBEANS. Verifica que las referencias internas a los paquetes sean correctas.
- c) Observa todas las clases e interfaces provistas en el Proyecto.
 - a. Son las utilizadas durante el curso para grafos dirigidos y no dirigidos
 - b. Observa con cuidado las interfaces **IGrafoContagio** e **IVerticeContagio** y las clases **TAnillosContagio** y **TGrafoContagios**.
- d) Observa el archivo principal "**Parcial.java**", contiene prácticamente todas las invocaciones que serán necesarias.
- e) La implementación requerida ha de cumplir con la interfaz entregada.
- f) Implementar las verificaciones básicas que aseguren que las funcionalidades anteriores se pueden ejecutar correctamente para el grafo representado.

PARTE 1: Funcionalidades a desarrollar

Dado un TDA **TGrafoContagios**, que hereda de **TGrafoNoDirigido** con las características y operaciones habituales, y en el cual has de representar **Personas** (la etiqueta es una string) y **contactos** (arcos que indican que existe probabilidad fundada de contagio entre cada par de personas), se requiere:

Siguiendo con la interfaz publicada, implementar un método de la clase **TGrafoContagios**, **anillosDeProbablesContagiados** que, recibiendo como parametros un **nombre de persona** (string) y un **entero** que indica la **distancia máxima de contacto** a considerar devuelve una instancia de la clase "**AnillosContagio**" (en la que, para cada distancia de contacto, se almacena el conjunto de personas correspondiente), para las **distancias 1 a maxDistancia**. Si **maxDistancia** es 0, se deben explorar **todos los contactos alcanzables** desde la persona indicada.

de **TGrafoContagios**

AnillosContagio anillosDeProbablesContagiados (String personaCOVID, int maxDistancia)

de TVertice

void obtenerAnillos(TAnillosContagio losAnillos , int maxDistancia)

NOTA IMPORTANTE: Observa que la clase **AnillosContagio** exhibe una operación **“agregarContagio”** que recibe como parámetros la distancia de contacto y nombre de la persona correspondiente - **agregarContagio(entero distanciaContacto, string nombrePersona)** -

PARTE 2: PROGRAMA PRINCIPAL

1. Implementa un programa principal que:
 - a. Cargue el Grafo con los Vértices contenidos en el archivo **“personas.txt”**.
 - b. Cargue las Aristas contenidas en el archivo **“contactos.txt”**.
 - c. Invoque la operación **“anillosDeProbablesContagiados”** del grafo, con los parámetros **indicados en el pizarrón (PersonaCOVID y maxDistancia)**.
 - d. Imprima por consola la cantidad de contactos que se encuentran a la distancia de contacto **indicada en el pizarrón (distAnilloBuscar)**.
 - e. Emita un archivo de salida **“anillos.txt”** con los listados de cada conjunto de personas que se encuentran a una **distancia de contagio menor o igual a la indicada** en el pizarrón (**“DistMaxParaArchivoListado”**). Este archivo debe tener la siguiente estructura:

1 LINEA INDICANDO “DISTANCIA de CONTACTO: <valor>”

seguida de los nombres de las personas en ese conjunto, ordenados alfabéticamente, indentados 2 espacios. Ejemplo:

DISTANCIA de CONTACTO: 1:

Nombre persona1

Nombre persona 2

DISTANCIA de CONTACTO: 2

Nombre persona 3

PARTE 3: TEST CASES

Desarrollar los **casos de prueba** (“test cases”) necesarios, completos y exhaustivos para verificar la corrección de los métodos implementados (sólo a nivel del TGrafoContagio).

ENTREGA

Subir a la webasignatura, en la tarea **“PARCIAL2-PARTE3”** el proyecto completo realizado, más el archivo de salida **“salida.txt”**.

ARCHIVOS DE ENTRADA

NOTA IMPORTANTE: LOS ARCHIVOS **PUEDEN** CONTENER ERRORES. DEBES IMPLEMENTAR LAS PRECAUCIONES BASICAS PARA EVITAR QUE EL SISTEMA COLAPSE ANTE ERRORES O INCONGRUENCIA DE DATOS.