

Ejercicio 1

Parte a)

/*Pre: Portador ya infectado como activo. Inoculado ya infectado como inmune*/

PROCEDIMIENTO CantidadSanos(comunidad : Grafo, portador : vertice, inoculado : vertice)

VAR contador : entero

COMIENZO

contador=0

propagacion_BFS(portador)

sanos_BFS(inoculado)

RETORNAR contador

FIN

PROCEDIMIENTO propagacion_BFS(portador : vertice)

VAR Cola Q

Vertice u

COMIENZO

marcar(portador)

insBack(a, Q)

Mientras Q no-vacia

u=primeroCola(Q)

desencolar(Q)

Para cada w adycente a u

Si w no marcado

Si w esInmnune marcar(w)

Si w esSano y u esActivo

setLatente(w)

marcar(w)

insBack(w)

Fin Si

Si w esSano y u esLatente

setActivo(w)

marcar(w)

insBack(w)

Fin Si

Fin si

Fin para

Fin Mientras

FIN

PROCEDIMIENTO sanos_BFS(inoculado: vertice)

VAR Cola Q

Vertice u

COMIENZO

insBack(inoculado)

Mientras Q no vacia

u=primeroCola(Q)

desencolar(Q)

Para cada w adyacente a u

Si w no marcado

marcar(w)

contador++

insBack(w)

Fin si

Fin Para

Fin Mientras

FIN

Parte b)

Todo camino que comience en un infectado y culmine en un sano debe contener entre sus v rtices al inmune (inoculado), en caso contrario la infecci n no se detendr a y continuar a el recorrido por la comunidad.

Por lo que el subgrafo obtenido al borrar el v rtice inmune contiene m  s componentes conexas que el original.

Es decir el v rtice inmune es un punto de articulaci n.

Parte c)

Realizar una recorrida para cada v rtice, salvo el Activo inicial, guardar la cantidad de sanos que quedar an para el v rtice que es inmunizado en cada pasada. Luego devolver al individuo que mayor de individuos sanos deje.

Parte d)

El grafo es conexo y no dirigido, se puede plantear la siguiente soluci n.

Hacer una recorrida DFS desde el activo.

Como el grafo es no dirigido no habr   aristas cross entre las ramas generadas del  rbol.

Contar para cada rama del  rbol generado por DFS la cantidad de v rtices que tienen.

Elegir para inocular el v rtice adyacente al activo en el  rbol DFS que tenga m  s v rtices en el camino hasta la hoja (sumidero).

Ejercicio 2)

Parte a)

Es posible solamente en el caso de que exista un camino simple que comunique a todos los activos y si existiese otro camino simple que comunique a todos los infectados latentes.

En otro caso existe al menos un infectado del otro estado que está actuando como punto de articulación y evite la propagación de la cura.

Parte b. i)

EsPeorEscenario(comunidad : Grafo) : Bool

VAR esPeor: Bool

COMIENZO

 esPeor=VERDADERO

 Para cada v que pertenece a V inicializar v como no marcado

 Para cada v

 Si v no marcado

 esPeor=PeorEscenario_BFS(v)

 Si esPeor== FALSO

 retornar esPeor

 Fin si

 Fin si

 Fin para

 retornar esPeor

FIN

PeorEscenario_BFS(v: vertice) : Bool

 VAR Q : Cola

 u : vertice

 esPeor : Bool

COMIENZO

 esPeor= VERDADERO

 marcar(v)

 insBack(v, Q)

 Mientras Q no-vacia

 u=primeroCola(Q)

 desencolar(Q)

 Para cada w adyacente a u

 Si estado w es igual a estado u

 esPeor=FALSO

 Fin si

 marcar(w)

insBack(w)

Fin para

Fin Mientras

RETORNAR esPeor

FIN

b.ii)

El sugrafo inducido por los v~~o~~rtices en estado latente y el inducido por los v~~o~~rtices en estado activo cada uno de ellos no tiene aristas.

propagacion_DFS(v: vertice, ik : vertice, lista : Lista de vertices, encontrado: bool)

COMIENZO

 marcar(v)

 insLista(v, lista)

 Para cada w adyacente a v

 Si w no marcado y w diferente de ik y encontrado == FALSO

 propagacion_DFS(w, ik, lista, b)

 borrarLista(v, lista)

 Sino si w igual a ik

 insLista(w, lista)

 encontrado=VERDADERO

 Fin Si

Fin para

FIN