



Ejercicio Nivel 11

Sistema Experto

Descripción global

La Facultad de Medicina de la Universidad de los Andes quiere construir un programa que sirva de apoyo a los profesionales de la salud para el diagnóstico y el tratamiento de enfermedades de sus pacientes. El programa permite guardar una serie de síntomas, a partir de los cuales los profesionales de la salud pueden determinar si un paciente padece de alguna enfermedad de las que están registradas en el sistema.

Objetivos

El objetivo de este ejercicio es que el estudiante comprenda y adquiera práctica en:

- El desarrollo de aplicaciones siguiendo un proceso incremental.
- El manejo de estructuras recursivas.
- El desarrollo de algoritmos recursivos.

Los siguientes pasos conforman el plan sugerido para desarrollar el ejercicio. La idea es ir desarrollando y probando incrementalmente los métodos de las clases. **No se preocupe si las clases de la interfaz o de las pruebas (test) tienen errores. Estos desaparecerán cuando termine (correctamente) los cambios en el modelo del mundo.**

Este ejercicio debe ser realizado de manera **INDIVIDUAL**.

Preparación

1. Descargue del sitio web del curso el archivo demo de la aplicación (del enlace llamado **n11_sistemaExperto_demo**) y ejecútelo para conocer el funcionamiento esperado del programa.
2. Descargue del sitio web del curso el esqueleto del ejercicio (del enlace llamado **n11_sistemaExperto_esqueleto**) del ejercicio del sitio del curso. Descomprima este archivo e importe el proyecto llamado **n11_sistemaExperto** en Eclipse.
3. Lea el enunciado del problema disponible en:
n11_sistemaExperto/docs/specs/Descripcion.pdf.
4. Estudie el documento de requerimientos funcionales disponible en:
n11_sistemaExperto/docs/specs/RequerimientosFuncionales.pdf.

5. Estudie el modelo del mundo diseñado para este ejercicio. Este modelo se encuentra en: **n11_sistemaExpertodocs/specs/ModeloConceptual.jpg**. Identifique las clases, relaciones entre clases, constantes, atributos y métodos
6. Asegúrese de tener activado el uso de aserciones para la ejecución del programa. Ver el tutorial en:
http://cupi2.uniandes.edu.co/sitio/images/cursosCupi2/apo2/tutoriales/n7_assert.pdf

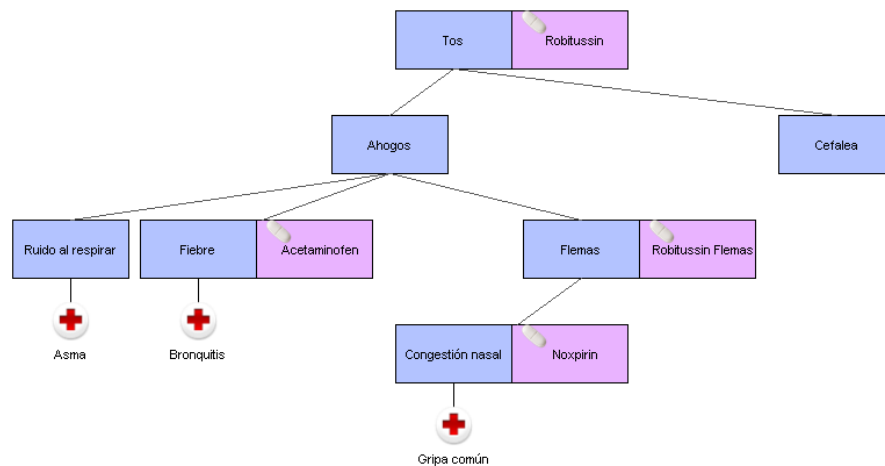
Parte1: Algorítmica de árboles - Completar la clase Síntoma

1. Complete el método `esHoja` de acuerdo con la documentación suministrada. Este método determina si el nodo del árbol es una hoja.
2. Complete el método `darAltura` de acuerdo con la documentación suministrada. Este método determina el número de síntomas que posee el subárbol del síntoma.
3. Complete el método `buscarSintoma` de acuerdo con la documentación suministrada. Este método busca un síntoma entre el síntoma actual y sus descendientes.
4. Complete el método `buscarEnfermedad` de acuerdo con la documentación suministrada. Este método busca una enfermedad dada en el síntoma actual o en sus descendientes.
5. Complete el método `eliminarSintoma` de acuerdo con la documentación suministrada, el cual elimina un síntoma con el nombre dado. Solo se puede eliminar un síntoma si es una hoja del árbol. El síntoma se puede eliminar aunque tenga asociada una enfermedad o un síntoma.
6. Complete el método `darEnfermedadesSintoma` de acuerdo con la documentación suministrada. Este método agrega todas las enfermedades del subárbol de este síntoma a una lista dada por parámetro.
7. Complete el método `darFarmacosEnfermedad` de acuerdo con la documentación suministrada. Este método agrega todos los fármacos asociados a los síntomas de una enfermedad (en la rama de la enfermedad).

Ejemplo:

Dado el siguiente árbol:





Los fármacos asociados a la enfermedad Gripa común serían:

1. Robitussin
2. Robitussin Flemas
3. Noxpirin.

Estrategia de solución sugerida:

- Para cada nodo del árbol, determinar si la enfermedad dada (en el ejemplo anterior Gripa común) se encuentra en su subárbol.
- Si en ese subárbol se encuentra la enfermedad y el nodo en cuestión tiene un fármaco asociado, este último se agrega a una lista (por medio de la técnica de acumulación de parámetros).
- Puede utilizar el método del punto 4 para determinar si una enfermedad se encuentra o no en un subárbol.

8. Complete el método `darListaSintomas` de acuerdo con la documentación suministrada. Este método recorre en preorden el árbol agregando los síntomas a la lista dada por parámetro.
9. Complete el método `darListaEnfermedades` de acuerdo con la documentación suministrada. Este método recorre en posorden el árbol agregando las enfermedades a lista dada por parámetro.
10. Complete el método `contarOcurrenciasNombreFarmaco` de acuerdo con la documentación suministrada. Este método cuenta el número de ocurrencias de un fármaco en el síntoma y en sus síntomas hijos.

Parte 2: Algorítmica de árboles - Completar la clase SistemaExperto

1. Complete el método `darPrimerSintoma` de acuerdo con la documentación suministrada, el cual devuelve el síntoma a la cabeza del árbol.



2. Complete el método `buscarSintoma` de acuerdo con la documentación suministrada, el cual busca el síntoma con el nombre dado por parámetro.
3. Complete el método `agregarSintoma` de acuerdo con la documentación suministrada. Este método agrega a un síntoma un síntoma hijo con la información dada por parámetro. No se agrega el síntoma si:
 - El síntoma padre no existe.
 - El síntoma padre tiene una enfermedad asociada.
 - Ya existe un síntoma registrado en el sistema con el nombre dado.Se debe verificar el invariante al final del método.
4. Complete el método `darEnfermedadesAsociadasSintoma` de acuerdo con la documentación suministrada. Este método agrega todas las enfermedades del subárbol del síntoma a la lista que entra por parámetro.
5. Complete el método `darFarmacosEnfermedad` de acuerdo con la documentación suministrada. Este método agrega todos los fármacos de la rama de una enfermedad a la lista que entra por parámetro.
6. Complete el método `eliminarSintoma` de acuerdo con la documentación suministrada. Este método elimina un síntoma del sistema. Si la raíz es nula o si el síntoma a eliminar no es una hoja no se elimina el síntoma.
7. Complete el método `asociarEnfermedad` de acuerdo con la documentación suministrada. Este método asocia una enfermedad a un síntoma cuyo nombre es dado por parámetro. No se asocia la enfermedad si:
 - El síntoma no es una hoja del árbol.
 - Ya existe una enfermedad con el mismo nombre dado por parámetro.
 - No existe un síntoma con el nombre dado por parámetro.
8. Complete el método `generarReporte` de acuerdo con la información suministrada en el contrato y el documento de requerimientos funcionales. Este método genera un reporte en la ruta y con el nombre especificado con el formato definido. En este se incluye la información de los síntomas, enfermedades y fármacos registrados en el sistema (en preorden). Si la carpeta o el archivo donde se quiere guardar el reporte no existe, en este método se tiene que crear.
9. Complete el método `contarOcuurrenciasNombreFarmaco` de acuerdo con la documentación suministrada. Este método cuenta el número de ocurrencias del nombre de un fármaco en el sistema.

Validación

Para comprobar el funcionamiento de su ejercicio usted puede:

1. Ejecutar las pruebas automáticas disponibles en el ejercicio (validación funcional del mundo) que le permitirán verificar la correcta implementación de sus métodos. Para



ejecutar las pruebas en el entorno de Eclipse presione click derecho sobre el paquete “uniandes.cupi2.editorUml.test” (o alguna de sus clases internas), seleccione la opción *Run as* y posteriormente la opción *JUnitTest*.

2. Ejecutar el programa e interactuar con todas las opciones disponibles en la interfaz.

Entrega

Este ejercicio debe ser realizado de manera **INDIVIDUAL**.

1. Construya el archivo entregable con el ejercicio completo. No olvide revisar que su entrega cumple con lo especificado en las normas del curso referentes a entregas de ejercicios. Consultar:
<http://cupi2.uniandes.edu.co/sitio/index.php/cursos/apo1/normas-administrativas>
2. Entregue el archivo del ejercicio vía SicuaPlus, de acuerdo con las normas, fecha y hora de entrega.

