Informe

Elegimos un hash para el guardado de doctores y pacientes, debido a que las busquedas del mismo son en O(1) y no necesitabamos un orden para los mismos. Se nos pasan por parametro inicialmente. Para el hash de doctores, la clave es el nombre del doctor y el dato es la especialidad del mismo. Para los pacientes, la clave es el nombre del paciente, y el dato el anio de ingreso.

Ambos hashes, utilizan la funcion free para destruir el dato, debido a que leyendo linea por linea, estamos liberando los datos de ambos hashes.

Luego, creamos un hash para las especialidades de los doctores, con clave, el nombre de la especialidad y dato un TDA que modela una lista de espera, desarrollado en el siguiente item.

Tambien creamos un ABB para los doctores que iban atendiendo a cada paciente, con clave el doctor, y dato el paciente atendido. La decision del uso del ABB fue debido a que a la hora de pedir el informe de la clinica, en el caso de ser un rango acotado de doctores, la busqueda debia ser en tiempo logaritmico.

TDA Espera

Elegimos crear nuestro propio TDA ya que necesitabamos poder manejar cierta prioridad en la espera de la clinica. La consigna nos decia que habia tanto pacientes urgentes, como regulares y los urgentes tenia mas prioridad a la hora de ser atentidos por los doctores.

Por lo tanto, para poder manejar esta prioridad a la hora de atenderlos, decidimos crear un TDA Espera, el cual consta de otros dos TDAs, una Cola y un ABB.

Para guardar a los pacientes urgentes, utlizamos una Cola ya que el orden relaticvo de la misma es del tipo FIFO (first in, first out). Esto es importante ya que los turnos uregentes se atienden por orden de llegada. Ademas, cumplia con la condicion del enunciado, la cual decia que atender un paciente urgente sea en O(1).

Para guardar a los pacientes regulares utilizamos un ABB que tiene como clave \*\*\* completar