

| | | | | | |
|--------------------|--|--------------|-------|--------------|--------------|
| EVALUACION | Obligatorio | GRUPO | TODOS | FECHA | Octubre 2020 |
| MATERIA | Algoritmos y Estructuras de Datos 1 | | | | |
| CARRERA | Analista Programador / Analista en Tecnologías de la información | | | | |
| CONDICIONES | <p>- Puntos: Máximo: 50 Mínimo: 1 - Fecha máxima de entrega : 26/11/2020 antes de las 21 hs</p> <p>LA ENTREGA SE REALIZA EN FORMA ONLINE EN ARCHIVO NO MAYOR A 40MB EN FORMATO ZIP, RAR O PDF.</p> <p>IMPORTANTE:</p> <ul style="list-style-type: none"> - Inscribirse - Formar grupos de hasta tres personas. - Subir el trabajo a Gestión antes de la hora indicada, ver hoja al final del documento: RECORDATORIO" | | | | |

Obligatorio: Sistema de Salud - Emergencia Covid

Contenido

| | | |
|-------|--|----|
| 1. | Introduction | 2 |
| 2. | Funcionalidades..... | 3 |
| 2.1. | Crear Sistema de Emergencia Covid | 3 |
| 2.2. | Destruir Sistema de Emergencia Covid | 3 |
| 2.3. | Registrar Ciudad..... | 3 |
| 2.4. | Registrar Hospital | 4 |
| 2.5. | Ingresar Servicio | 4 |
| 2.6. | Borrar Servicio | 5 |
| 2.7. | Ingresar Comentario | 5 |
| 2.8. | Realizar Ingreso..... | 6 |
| 2.9. | Cancelar Ingreso | 6 |
| 2.10. | Listado de Servicios | 7 |
| 2.11. | Listado de Hospitales por Ciudad | 8 |
| 2.12. | Listado de Hospitales por Ranking | 8 |
| 2.13. | Listado de Comentarios..... | 9 |
| 2.14. | Listado de Espera..... | 9 |
| 3. | Ejercicio complementario 1 | 10 |
| 4. | Ejercicio complementario 2 | 10 |
| 4.1. | Cargar Matriz de Distancias..... | 10 |
| 4.2. | Camino más corto | 10 |

1. Introduction

Se desea implementar un sistema para la gestión de la internación de pacientes en hospitales pertenecientes al sistema de salud.

Los hospitales serán registrados en el sistema de salud - emergenciacoVID con la ciudad a la que pertenecen, su nombre (que podrá ser único dentro de la ciudad), una cierta cantidad de habitaciones/salas y camas (capacidad), la categoría (cantidad de estrellas), y una lista de servicios provistos.

Para simplificar la realidad consideraremos solo las salas de los hospitales unitarias, o sea que cada habitación podrá ser ocupada por un único paciente contemplando el protocolo de distanciamiento establecido.

El sistema deberá proveer funcionalidades para que los pacientes puedan gestionar su internación en un hospital, realizar búsquedas por ciudad o por ranking, consultar los servicios de los mismos, etc.

En **una segunda etapa** se pretende llevar con mayor detalle de la internación de los pacientes registrando la sala de ingreso y si es con o sin respirador. También distinguir entre salas con un solo paciente y salas con más de un paciente internados por otros motivos que no requieren distanciamiento.

Se proveen los siguientes tipos de datos que deberán ser respetados.

| | |
|----------------|---|
| Sistema | <pre>public class Retorno { enum Resultado {OK, ERROR, NO_IMPLEMENTADA}; int valorEntero; String valorString; boolean valorBooleano; Resultado resultado; /*Aquí introduzca la información que estime conveniente*/ }</pre> |
|----------------|---|

Pueden definirse tipos de datos (clases) auxiliares.

2. Funcionalidades

2.1. Crear Sistema de Emergencia Covid

Firma: RetornocrearSistemaIngreso();

Descripción: Crea la estructura necesaria para representar el Sistema de Emergencia Covid.

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> Siempre retorna OK. |
| ERROR | <input type="checkbox"/> No existen errores posibles. |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

Nota: el sistema debe recibir como parámetro la cantidad de ciudades que va a gestionar, si el parámetro es 0 no se limita esta cantidad, de lo contrario se valida la misma.

2.2. Destruir Sistema de Emergencia Covid

Firma: RetornodestruirSistemaIngreso();

Descripción: Destruye el Sistema de Emergencia Covid y todos sus elementos, liberando la memoria utilizada.

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> Siempre retorna OK. |
| ERROR | <input type="checkbox"/> No existen errores posibles. |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.3. Registrar Ciudad

Firma: RetornoregistrarCiudad(String Ciudad);

Descripción: Registra la Ciudad "Ciudad"

Nota: esto se utilizará para la parte 2 del obligatorio para definir la matriz de ciudades (mapa de ciudades)

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> En caso que la ciudad "Ciudad" haya sido ingresado correctamente en el sistema. |
| ERROR | <input type="checkbox"/> 1. En caso que la ciudad "Ciudad" ya existe en el sistema. |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.4.Registrar Hospital

Firma: RetornoregistrarHospital(String Ciudad, String Nombre, intEstrellas,int Capacidad, int ranking);

Descripción: Registra el Hospital de nombre “Nombre”, en la ciudad “Ciudad”, con cantidad de estrellas “Estrellas”, capacidad “Capacidad” y ranking 0 en el Sistema de Emergencia Covid.

Nota: Al inicio los Hospitales registrados no contarán con ningún servicio.

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> En caso que el Hospital haya sido ingresado correctamente en el sistema. |
| ERROR | <input type="checkbox"/> 1. En caso que “Estrellas” sea menor a 1 o mayor a 5. |
| | <input type="checkbox"/> 2. En caso que “Capacidad” sea menor a 0. |
| | <input type="checkbox"/> 3. En caso que ya exista un Hospital de nombre “Nombre” dentro de la ciudad “Ciudad”. |
| | <input type="checkbox"/> 4. En caso que no exista la ciudad “Ciudad” |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.5.Ingresar Servicio

Firma: RetornoingresarServicio(String Ciudad, StringHospital, String Servicio);

Descripción: Ingresa el servicio “Servicio” al Hospital “Hospital”.

Nota 1: No se hará ningún control de unicidad sobre los servicios.

Nota 2: El ingreso de los servicios deberá realizarse en el menor tiempo posible.

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> En caso que el servicio “Servicio” haya sido ingresado correctamente en el Hospital “Hospital”. |
| ERROR | <input type="checkbox"/> 1. En caso que no exista un Hospital “Hospital” registrado dentro de la ciudad “Ciudad”. |
| | <input type="checkbox"/> 2. En caso que no exista la ciudad “Ciudad”. |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.6.Borrar Servicio

Firma: RetornoborrarServicio(String Ciudad, StringHospital, String Servicio);

Descripción: Borra la primera ocurrencia del servicio "Servicio" del Hospital "Hospital".

| Retornos posibles | |
|------------------------|---|
| OK | <input type="checkbox"/> En caso de que el servicio "Servicio" haya sido borrado correctamente del Hospital "Hospital". |
| ERROR | <input type="checkbox"/> 1. En caso de que no exista un Hospital "Hospital" registrado dentro de la ciudad "Ciudad". |
| | <input type="checkbox"/> 2. En caso de que no exista el servicio "Servicio" registrado en el Hospital "Hospital". |
| | <input type="checkbox"/> 3. En caso que no exista la ciudad "Ciudad". |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.7.Ingresar Comentario

Firma: RetornoingresarComentario(String Ciudad, String Hospital, String Comentario, int Ranking);

Descripción: Ingresa el comentario "Comentario" al Hospital "Hospital". Cada comentario llevará una calificación para el Hospital de acuerdo a su nivel de atención.

El ranking general del Hospital quedará definido por el promedio de todas sus calificaciones.

Nota: El ranking será representado con un número entre 0 y 5 inclusive.

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> En caso de que el comentario "Comentario" haya sido ingresadocorrectamente en el Hospital "Hospital". |
| ERROR | <input type="checkbox"/> 1. En caso que el ranking sea menor a 0 o mayor a 5. |
| | <input type="checkbox"/> 2. En caso que no exista un Hospital de nombre "Hospital" registrado dentro de la ciudad "Ciudad". |
| | <input type="checkbox"/> 3. En caso que no exista la ciudad "Ciudad". |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.8. Realizar Ingreso

Firma: RetornorealizarIngreso(intPaciente, String Ciudad, StringHospital);

Descripción: Realiza Ingreso de un paciente a una habitación en el Hospital "Hospital" dentro de la ciudad "Ciudad". En caso de que no haya cupos de habitaciones/camas disponibles, el Paciente "Paciente" quedará en lista de espera.

| Retornos posibles | |
|------------------------|---|
| OK | <input type="checkbox"/> En caso de que la reserva haya sido efectuada correctamente en el Hospital "Hospital". |
| | <input type="checkbox"/> En caso de que la reserva haya quedado en lista de espera. |
| ERROR | <input type="checkbox"/> 1. En caso de que no exista un Hospital de nombre "Hospital" registrado dentro de la ciudad Ciudad". |
| | <input type="checkbox"/> 2. En caso que no exista la ciudad "Ciudad". |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.9. Cancelar Ingreso

Firma: RetornocancelarIngreso(intPaciente, String Ciudad, StringHospital);

Descripción: Cancela el Ingreso en el Hospital "Hospital" para el Paciente "Paciente". En caso de que la cancelación se lleve a cabo correctamente y haya Pacientes en lista de espera, el cupo liberado será ocupado por el primer Paciente de la lista.

La búsqueda de los Ingresos se efectuará primero dentro de las Ingresos en habitaciones/camas y luego en la lista de espera.

| Retornos posibles | |
|------------------------|---|
| OK | <input type="checkbox"/> En caso que la cancelación del Ingreso se lleve a cabo correctamente en el Hospital "Hospital". |
| ERROR | <input type="checkbox"/> 1. En caso de que no exista un Hospital de nombre "Hospital" registrado en la ciudad "Ciudad". |
| | <input type="checkbox"/> 2. En caso de que el Paciente "Paciente" no tenga ningún Ingreso en el Hospital "Hospital" registrado en la ciudad "Ciudad". |
| | <input type="checkbox"/> 3. En caso que no exista la ciudad "Ciudad". |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.10. Trasladar Paciente

Firma: RetornotrasladarPaciente(intPaciente, StringCiudadOrigen, StringHospitalOrigen,CiudadDestino, StringHospitalDestino);

Descripción: Traslada el paciente "Paciente" del Hospital "HospitalOrigen" al Hospital "HospitalDestino". En caso de que el traslado se lleve a cabo correctamente y haya Pacientes en lista de espera, el cupo liberado será ocupado por el primer Paciente de la lista.

| Retornos posibles | |
|------------------------|---|
| OK | <input type="checkbox"/> En caso que el traslado se lleve a cabo correctamente. |
| ERROR | <input type="checkbox"/> 1. En caso de que no exista un Hospital de nombre "HospitalOrigen" registrado en la ciudad "CiudadOrigen". |
| | <input type="checkbox"/> 2. En caso de que no exista un Hospital de nombre "HospitalDestino" registrado en la ciudad "CiudadDestino". |
| | <input type="checkbox"/> 3. En caso de que el Paciente "Paciente" no tenga ningún Ingreso en el Hospital "HospitalOrigen" registrado en la ciudad "Ciudad". |
| | <input type="checkbox"/> 4. En caso que no exista la ciudad "CiudadOrigen". |
| | <input type="checkbox"/> 4. En caso que no exista la ciudad "CiudadDestino". |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.11. Listado de Servicios

Firma: RetornolistarServicios(String Ciudad, StringHospital);

Descripción: Lista todos los servicios prestados por el Hospital "Hospital" de la ciudad "Ciudad" con el siguiente Formato:

Formato.

Servicios del Hospital<Hospital><Ciudad>

1 - <Servicio1>

...

N - <ServicioN>

En caso que no haya servicios registrados en el Hospital "Hospital", el sistema deberá imprimir: No existen servicios registrados en el Hospital<Hospital><Ciudad>

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> En caso que se imprima el listado correctamente. |
| ERROR | <input type="checkbox"/> 1. En caso que no exista el Hospital "Hospital" registrado en la ciudad "Ciudad". |
| | <input type="checkbox"/> 2. En caso que no exista la ciudad "Ciudad". |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.12. Listado de Hospitales por Ciudad

Firma: RetornolistarHospitalesCiudad(String Ciudad);

Descripción: Lista todos los Hospitales registrados en la ciudad "Ciudad" ordenados por Nombre con el siguiente

formato.

Hospitales en <Ciudad>

<NombreHospital1><Estrellas1><Ranking1>

...

<NombreHospitalN><EstrellasN><RankingN>

En caso que no haya ningún Hospital registrado en la ciudad "Ciudad" el sistema deberá imprimir: No existen Hospitales registrados en <Ciudad>

Nota: La impresión de este listado, por ser muy utilizado, deberá realizarse en el menor tiempo posible.

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> En caso que se imprima el listado correctamente. |
| ERROR | <input type="checkbox"/> 1. En caso que no exista la ciudad "Ciudad". |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.13. Listado de Hospitales por Ranking

Firma: RetornolistarHospitalesRanking();

Descripción: Lista todos los Hospitales registrados en el sistema ordenados por ranking descendente.

El formato de impresión deberá ser el siguiente:

Hospitales ordenados por ranking

<Ciudad1> - <Hospital1> - <Ranking1>

...

<CiudadN> - <HospitalN> - <RankingN>

En caso que no exista ningún Hospital registrado en el sistema, se deberá imprimir: No hay registros de Hospitales en el sistema.

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> Siempre retorna OK. |
| ERROR | <input type="checkbox"/> No existen errores posibles. |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.14. Listado de Comentarios

Firma: RetornolistarComentarios(String Ciudad, StringHospital);

Descripción: Lista todos comentarios ingresados para el Hospital "Hospital" dentro de la ciudad "Ciudad". Los comentarios más recientes deberán aparecer primeros en el listado.

El formato deberá ser el siguiente:

N - <Comentario N> - <RankingN>

...

1 - <Comentario 1> - <Ranking1>

En caso que no existan comentarios para el Hospital "Hospital" el sistema deberá imprimir:

No se han agregado comentarios al Hospital<Hospital><Ciudad>

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> En caso que se imprima el listado correctamente. |
| ERROR | <input type="checkbox"/> 1. En caso que no exista el Hospital "Hospital" registrado en la ciudad "Ciudad". |
| | <input type="checkbox"/> 2. En caso que no exista la ciudad "Ciudad". |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

2.15. Listado de Espera

Firma: RetornolistarEspera(String Ciudad, StringHospital);

Descripción: Lista los Pacientes en lista de espera para el Hospital "Hospital" dentro de la ciudad "Ciudad". Los Pacientes deberán imprimirse en el orden que serán considerados para tomar una posible habitación libre. Se considerará primero al Paciente que esté hace más tiempo en la lista de espera.

El formato del listado deberá ser el siguiente:

Lista de espera para el Hospital<Hospital><Ciudad>

1 - <CedulaPaciente1>

...

N - <CedulaPacienteN>

En caso que no existan Pacientes en la lista de espera se deberá imprimir en pantalla: No existen Ingreso pendientes para el Hospital<Hospital><Ciudad>

| Retornos posibles | |
|------------------------|--|
| OK | <input type="checkbox"/> En caso que se imprima el listado correctamente. |
| ERROR | <input type="checkbox"/> 1. En caso que no exista el Hospital "Hospital" registrado en la ciudad "Ciudad". |
| | <input type="checkbox"/> 2. En caso que no exista la ciudad "Ciudad". |
| NO_IMPLEMENTADA | <input type="checkbox"/> Cuando aún no se implementó. Es el tipo de retorno por defecto. |

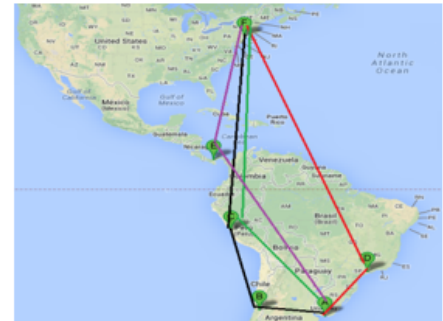
3. Ejercicio segunda etapa

Indique los cambios necesarios en la estructura y las funcionalidades que se requieren implementar (indique las firmas necesarias solamente) para cumplir con la segunda etapa

4. Ejercicio complementario 2

Dada la siguiente matriz en la que se almacena las distancias que existen entre las ciudades.

| | A Montevideo | B Santiago | C Lima | D San Pablo | E Panamá | F New York |
|---|-----------------|---------------|-----------|-------------------|-------------|---------------|
| A | 0 | 10 | 15 | 30 | 0 | |
| B | 10 | 0 | 20 | 0 | 0 | |
| C | 25 | 20 | 0 | 0 | 40 | |
| D | 15 | 0 | 0 | 0 | 0 | 45 |
| E | 30 | 0 | 0 | 0 | 0 | 25 |
| F | 0 | 0 | 40 | 45 | 25 | 0 |



Montevideo, Lima, New York

Montevideo, Santiago, New York

Montevideo, San Pablo, New York

Montevideo, Panamá, New York

Se desea implementar los siguientes métodos:

4.1.Cargar Matriz de Distancias

Firma: RetornoCargarDistancias(int[][] Ciudades)

Nota: La matriz debe ser creada en función de la cantidad de ciudades definidas en el sistema. Debemos definir en la creación del sistema un parámetro cantidad de ciudades para luego validar y crear la matriz en función de esta cantidad.

4.2.Camino más corto

Firma: RetornoBuscarCamino(int [][] M, string origen, string destino)

Retorna una lista A → D → F →

Se debe retornar el camino más corto para llegar de un origen a un destino restringiendo la búsqueda a caminos que solo tengan una ciudad intermedia.

Nota: Las casillas que tienen valor distinto de 0 son ciudades que tienen conexión y las casillas que tienen valor 0 no tienen conexión.

Defina las estructuras que considere necesarias para resolver el problema.

Información importante

- ☐ Lectura de Obligatorio: _____.
- ☐ Puntaje mínimo/máximo: 1/50
- ☐ Los obligatorios se realizan por equipos de hasta 3 **estudiantes (por ser un periodo excepcional)**.
- ☐ Plazo máximo de segunda entrega (con boleta): _____.
- ☐ Se deberán respetar los formatos de impresión dados para las operaciones que imprimen en consola.
- ☐ El resto de las operaciones no deben imprimir nada en consola.
- ☐ El sistema no debe requerir ningún tipo de interacción con el usuario por consola.
- ☐ Es obligación del estudiante mantenerse al tanto de las aclaraciones/cambios que se realicen en clase o a través del foro de aulas.
- ☐ **Se valorará la selección adecuada de las estructuras para modelar el problema y la eficiencia encada una de las operaciones.**
- ☐ Deberá aplicar la metodología vista en el curso.
- ☐ Deberá incluir en la documentación: **La las pre y post condiciones de los métodos solicitados, un diagrama de la estructura de datos que se implementó para representar el Sistema de Emergencia Covid junto con una breve explicación indicando por qué eligió dichas estructuras y las pruebas realizadas con el resultado obtenido.**
- ☐ Para la presentación de la documentación se publicará en aulas.ort.edu.uy un template. (El uso de este template es obligatorio).
- ☐ El proyecto será implementado en lenguaje JAVA sobre una interfaz que se publicará en el sitio de la materia en aulas.ort.edu.uy (El uso de esta interfaz es obligatorio).

RECORDATORIO: IMPORTANTE PARA LA ENTREGA

- **Obligatorios**

La entrega de los obligatorios será en formato digital online, a excepción de algunas materias que se entregarán en Bedelía y en ese caso recibirá información específica en el dictado de la misma.

Los principales aspectos a destacar sobre la **entrega online de obligatorios** son:

1. Ingresá al sistema de Gestión.
2. En el menú, seleccioná el ítem "Evaluaciones" y la instancia de evaluación correspondiente, que figura bajo el título "Inscripto".
3. Para iniciar la entrega hacé clic en el ícono:
4. Ingresá el número de estudiante de cada uno de los integrantes y hacé clic en "Agregar". El sistema confirmará que los integrantes estén inscriptos al obligatorio y, de ser así, mostrará el nombre y la fotografía de cada uno de ellos. Una vez agregados todos los integrantes, hacé clic en "Crear equipo".

Cualquier integrante podrá:

- **Modificar la integración del equipo.**
- **Subir el archivo de la entrega.**

5. Seleccioná el archivo que deseás entregar. Verificá el nombre del archivo que aparecerá en la pantalla y hacé clic en "Subir" para iniciar la entrega. Cada equipo (hasta 2 estudiantes) debe entregar **un único archivo en formato zip o rar** (los documentos de texto deben ser pdf, y deben ir dentro del zip o rar). El archivo a subir debe tener **un tamaño máximo de 40mb**

Quando el archivo quede subido, se mostrará el nombre generado por el sistema (1), el tamaño y la fecha en que fue subido.

6. El sistema enviará un e-mail a todos los integrantes del equipo informando los detalles del archivo entregado y confirmando que la entrega fue realizada correctamente.
7. Podés cerrar la pestaña de entrega y continuar utilizando Gestión o salir del sistema.
8. La **hora tope para subir el archivo será las 21:00** del día fijado para la entrega.
9. La entrega se podrá realizar desde cualquier lugar (ej. hogar del estudiante, laboratorios de la Universidad, etc).
10. Aquellos de ustedes que presenten alguna dificultad con su inscripción o tengan inconvenientes técnicos, por favor contactarse con el Coordinador o Coordinación adjunta **antes de las 20:00hs.** del día de la entrega, a través de los mails gervaz@ort.edu.uy, alamon@ort.edu.uy y terra@ort.edu.uy, o telefónicamente al 29021505 - int 1156 (de 8:00 a 14:00 hs) y 1436 (de 17:30 a 20:00 hs).

Si tuvieras una situación particular de fuerza mayor, debes contactarte con suficiente antelación al plazo de entrega, al Coordinador de Cursos (gervaz@ort.edu.uy) o Secretario Docente (paulos@ort.edu.uy).