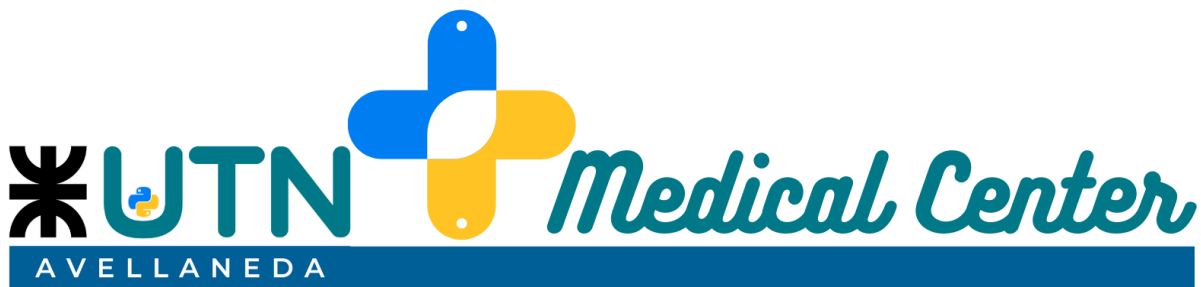


Universidad Tecnológica Nacional Facultad Regional Avellaneda										 Tecnatura Universitaria en Programación				
Materia:	Laboratorio I													
Pertenece a:	1º Cuatrimestre													
Apellido ⁽¹⁾ :						Fecha:	11 Julio de 2024							
Nombre/s ⁽¹⁾ :						Docente a cargo ⁽²⁾ :								
División ⁽¹⁾ :						Nota ⁽²⁾ :								
DNI ⁽¹⁾ :						Firma ⁽²⁾ :								
Instancia ⁽²⁾⁽³⁾ :	P1		RP1		P2	x	RP2		RIN		F			

(1) Campos a completar solo por el alumno.

(2) Campos a completar solo por el docente.

(3) Las instancias válidas son: 1º Parcial (**P1**), Recuperatorio de 1º Parcial (**RP1**), 2º Parcial (**P2**), Recuperatorio de 2º Parcial (**RP2**), Recuperatorio Integrador (**RIN**), Final (**F**), Recuperatorio de Final (**RF** - Solo válido para seminario de nivelación). Marcar lo que corresponda con una cruz.



La Clínica 'UTN-Medical Center' nos ha solicitado desarrollar un software de gestión de pacientes para llevar a cabo un control exhaustivo de los turnos.

Datos correspondientes a la clase Paciente:

- id (autoincremental e irrepetible)
- nombre
- apellido
- dni (irrepetible)
- edad
- fecha de registro
- obra social

Datos correspondientes a la clase Turno:

- id (autoincremental e irrepetible)
- id_paciente (debe existir en el sistema)
- especialidad
- monto a pagar (tener en cuenta el precio base y el descuento/recargo según la obra social del paciente)
- estado del turno

Datos correspondientes a la clase Clínica:

- Razón social (string)
- lista pacientes (lista)
- lista turnos (lista)
- especialidades (diccionario)
- obras sociales válidas (diccionario)
- recaudacion (float)
- hay_pacientes_sin_atencion (booleano)

Consideraciones:

- Al iniciar el programa se cargará en el sistema un JSON (**configs.json**) con las especialidades disponibles y obras sociales que acepta la clínica. Si hay pacientes cargados, se tendrán que cargar en el sistema también.
- El programa deberá gestionar turnos y atenciones de pacientes en una importante clínica. Para ello el programa deberá poder cargar en sistema al paciente, luego asignarle al paciente un turno con una especialidad entre:
 - Odontología
 - Médico Clínico
 - Psicología
 - Traumatología
- Cuando se registra un nuevo paciente en el sistema, se deberá buscar en la lista de pacientes que no exista otro con el mismo dni.
 - En caso de existir, mostrar un mensaje de error.
 - En caso de no existir, se debe dar de alta.
- Cuando se carga el turno, se asigna un paciente existente de la lista de pacientes y el estado por defecto será "Activo".
- Luego de ser atendido, el estado del turno cambiará a "Finalizado".
- Luego de ser finalizado se procederá a cobrar la atención, con lo cual el estado del turno cambiará nuevamente a "Pagado".

Validaciones:

- Nombre y Apellido del paciente: Debe contener solo caracteres alfabéticos y no exceder los 30 caracteres cada uno. No pueden contener números ni caracteres especiales.
- Edad: Debe ser un valor numérico entero entre 18 y 90.
- Obra social: Debe ser un valor entre: Swiss Médical, Apres, PAMI, Particular.
 - Si el paciente tiene 60 años o más, la única opción disponible a seleccionar será PAMI. Si tiene menos de 60, puede seleccionar las restantes EXCEPTO PAMI.
- Especialidad: El valor debe estar entre las opciones: Médico Clínico, Odontología, Psicología y Traumatología.
- Estado del turno: al momento de cargarlo, el estado por defecto será: "Activo".
- Monto a pagar (del turno): el monto se calculará teniendo en cuenta el precio base de la atención (**\$4000**) y luego aplicarle el descuento o recargo según la obra social del paciente. Además, a ese monto resultante, tener en cuenta lo siguiente:
 - Si el paciente tiene Swiss Medical (Aplicar -40% y además...):
 - Si la edad está entre 18 y 60, se aplica un -10% extra
 - Si el paciente tiene Apres (Aplicar -25% y además...):
 - Si la edad está entre 26 y 59, se aplica un -3% extra
 - Si el paciente tiene PAMI (Aplicar -60% y además...):
 - Si la edad es 80 o superior, se aplica un -3% extra
 - Si el paciente es Particular (Aplicar +5% y además...):
 - Si la edad está entre 40 y 60, se aplica un +15% extra

Opciones del menú:

1. Alta paciente: se dará de alta un paciente (si es que no existe ya en el sistema, validar por id y dni). Luego se deberá guardar el paciente en el JSON donde deberán prevalecer todos los pacientes (lista de pacientes) que fueron dados de alta en el sistema.
2. Alta turno: se dará de alta un turno para una especialidad, teniendo en cuenta el id de un paciente ya existente en el sistema. Luego ese turno se escribirá en el archivo JSON donde deben prevalecer todos los turnos creados (lista de turnos).
3. Ordenar turnos: Se ordenará la lista de turnos según opciones
 - a. Obra Social ASC
 - b. * Monto DESC
4. Mostrar pacientes en espera: Se deberá mostrar todos los pacientes que aún tienen el turno en "Activo".
5. Atender pacientes: Se seleccionará los primeros 2 pacientes cuyo estado del turno sea "Activo" y se cambiará el estado a "Finalizado". En caso de haber un solo paciente, se seleccionará ese mismo para ser atendido. En caso de no haber ninguno en espera, se informará en consola, mensaje a elección.
6. Cobrar atenciones: Se seleccionará a los pacientes cuyo estado del turno sea "Finalizado" para realizar el cobro de dicha atención (Tener en cuenta el valor correspondiente según obra social y edades). Se cambiará el estado del turno a "Pagado". Luego ese monto se sumará al tributo recaudación de la clase Clínica.
7. Cerrar caja: Solo cuando no haya más pacientes por atender (estado "Activo" o "Finalizado") se mostrará el total de lo recaudado y se actualizarán los archivos de pacientes y turnos, caso contrario se mostrará un mensaje que indique que aún hay pacientes por atender.
8. Mostrar informe: Se mostrará el informe determinado por el último dígito de tu DNI. Seleccioná **UNO Y NADA MÁS QUE UN SOLO INFORME** según el último dígito de tu DNI:
 - a. 0 - Informar el monto total obtenido en las atenciones
 - b. 1 - Informar cuántos pacientes fueron atendidos en total
 - c. 2 - Informar el monto obtenido por cada obra social
 - d. 3 - Informar con qué obra social se obtuvo más ingresos
 - e. 4 - Informar con qué obra social se obtuvo menos ingresos
 - f. 5 - Informar el número de pacientes atendidos por PAMI
 - g. 6 - Informar la especialidad más solicitada por los pacientes
 - h. 7 - Informar la especialidad menos solicitada por los pacientes
 - i. 8 - Informar todos los datos del paciente más joven atendido
 - j. 9 - Informar el promedio de edades de pacientes atendidos en una especialidad (a elección)
9. Salir: Cerrará el programa.

Requisitos adicionales:

- El programa deberá estar correctamente modularizado, haciendo uso de módulos, paquetes y funciones propias para solicitar enteros, flotantes y cadenas, así como para las validaciones de cada uno de estos tipos de datos.
- El código debe estar programado de manera eficiente y siguiendo buenas prácticas de la programación y las reglas de estilo de la cátedra incluyendo la documentación de las funciones.

Temas que deben aplicarse:

- Estructuras de control
- Bucles
- Funciones (Documentadas)
- Ordenamiento
- Listas
- POO (4 pilares)
- Lambdas
- Filter
- Map