

Prueba Sumativa N° 2

Programación Orientada a Objetos – 09 de Diciembre 2023

Nombre		RUT	
Paralelo	() APaulini () ERoss		

Antecedentes generales:

Puntaje total de la prueba/Puntos para nota aprobatoria(4.0)	100 puntos 60 puntos	Puntaje Obtenido	
Duración de la prueba	3 horas	Nota final	
Resultados de Aprendizaje a evaluar	1. Aplicar técnicas de ingeniería de software en la creación de software legible, mantenible y testeable. 2. Aplicar técnicas de programación orientada al objeto en la resolución de problemas. 3. Crear tipos de datos abstractos con bajo acoplamiento entre la implementación y su comportamiento que permitan la resolución problemas. 4. Analizar las relaciones causa efecto de los procesos en estudio. 6. Seleccionar los procesos, técnicas y herramientas adecuados de acuerdo a los requerimientos.		
Fecha de entrega de resultados	16 diciembre 2023		

Instrucciones:

1. Esta evaluación tiene 4 páginas (incluyendo la portada). Compruebe que dispone de todas las páginas.
2. Lea la prueba completamente **DOS** veces antes de hacer cualquier pregunta
3. Durante la prueba no se puede utilizar: teléfono móvil, apuntes. Está prohibido intentar conectarse a internet de cualquier manera (excepto a CampusVirtual, y solo para subir su solución). Si es sorprendido obtendrá la calificación mínima. Tampoco puede utilizar dispositivos de almacenamiento externos o cualquier otro dispositivo relojes inteligentes, ábacos, etc.
4. Una prueba respondida correctamente en un 60%, de acuerdo a las ponderaciones asignadas, corresponde a una nota 4,0.
5. La prueba es individual, cualquier sospecha de copia será calificada con la nota mínima y el caso será remitido al comité de ética.
6. En su espacio personal no debe haber nada más que hojas de papel en blanco, lápiz, goma.
7. El resto de sus implementos debe guardarlos dentro de su mochila/bolso y ésta debe posicionarse al frente debajo de la pizarra. Si leyó hasta este punto, felicidades, para saber que lo hizo dibuje una estrella al final de esta página.
8. Los estudiantes quienes se les compruebe falta de honestidad académica o cualquier otro acto contrario a las normas de permanencia universitaria o al espíritu universitario, serán sancionados, según sea la gravedad de la falta, con medidas desde la amonestación verbal hasta la suspensión o pérdida de la condición de estudiante, los estudiantes expulsados no podrán volver a ingresar a ninguna carrera, programa o curso de la institución. El estudiante que incurriere en falta de honestidad, durante la realización de un proceso evaluativo, será calificado con la nota mínima 1,0.

Acepto las condiciones firmando: _____

Problema 1. Property Boss (50 puntos)

Un corredor de propiedades, que comenzó arrendando las casas de sus amigos y familia, ha crecido tanto que ahora ofrece sus servicios de corretaje en diferentes localidades, a variados tipos de clientes y para diferentes tipos de bienes raíces.

En general todos los bienes raíces tienen:

- Dirección conformada por **Calle, Número y Comuna**.
- Avalúo Fiscal.

Los tipos de bienes raíces con los que trabaja serán enumerados a continuación, junto a sus principales características que los diferencian entre sí:

1. Casa Independiente: Costo **Aseo Municipal**
2. Casa en Condominio: Su dirección incluye **Letra** identificadora de la casa, ya que el resto de la dirección es común para el condominio. También incluye el valor de los **Gastos Comunes**.
3. Casa de Uso Comercial: **Tipo de Patente**, la que puede ser Comercial, Profesional o de Alcoholes.
4. Departamento: A la dirección se agrega **Número** de departamento y **Letra** de torre.
5. Espacio Comercial: A la dirección se agrega **Número** de local. **Tipo de Patente**, la que puede ser Comercial, Profesional o de Alcoholes.

El corredor necesita hacer un sistema que le entregue los ingresos percibidos si todos sus clientes arriendan en modalidad sin o con administración y separadas por tipo de bien raíz. Esto debe considerar solo un mes.

Contrato de arriendo mensual sin administración

Tipo	Arriendo Mensual	Comisión
Casa Independiente	El arriendo anual será del 6% del avalúo fiscal. Al valor mensual se le suman 3 veces el costo de Aseo Municipal.	50% del valor de arriendo mensual
Casa en Condominio	El arriendo anual será del 5% del avalúo fiscal. Al valor mensual se le suman los gastos comunes.	60% del valor de arriendo mensual
Casa de Uso Comercial	El arriendo anual será del 2% del avalúo fiscal. Al valor mensual se le suma el 30% del valor mensual si la patente es de Alcoholes.	70% del valor de arriendo mensual
Departamento	El arriendo anual será de 3.5% + 20% del valor de arriendo si el departamento se encuentra en un piso 5 o superior.	30% del valor de arriendo mensual
Espacio Comercial	El arriendo anual será del 3% del avalúo fiscal. Al valor mensual se le suma el 25% del valor mensual si la patente es Profesional o Comercial.	80% del valor de arriendo mensual

Contrato de arriendo mensual con administración

Tipo	Arriendo Mensual	Comisión
Casa Independiente	El arriendo anual será del 5.5% del avalúo fiscal. Al valor mensual se le suman 3 veces el costo de Aseo Municipal.	15%
Casa en Condominio	El arriendo anual será del 4.8% del avalúo fiscal. Al valor mensual se le suman los gastos comunes.	15%
Casa de Uso Comercial	El arriendo anual será del 2% del avalúo fiscal. Al valor mensual se le suma el 30% del valor mensual si la patente es de Alcoholes.	20%
Departamento	El arriendo anual será de 3.5% + 20% del valor de arriendo si el departamento se encuentra en un piso 5 o superior.	10%
Espacio Comercial	El arriendo anual será del 3% del avalúo fiscal. Al valor mensual se le suma el 25% del valor mensual si la patente es Profesional o Comercial.	20%

El archivo propiedades.txt tiene en cada línea una propiedad, el archivo tiene la siguiente estructura:

Tipo_propiedad,Calle,Número,Comuna,Avalúo_fiscal,Atributo_diferencial1,[Atributo_diferencial2]

Los atributos diferenciales de cada propiedad vendrán en el mismo orden que aparecen nombrados anteriormente, pudiendo ser 1 o 2 según el tipo.

Debe entregar:

- Código Java (100%)

Consideraciones:

- Todos los departamentos utilizan la nomenclatura PXX, donde P indica el número de piso y XX indica el número de departamento específico de ese piso. Por ejemplo, el 404 sería el departamento 4 del piso 4.
- Redondee los resultados finales a la unidad.
- Debe usar orientación al objeto. Debe usar herencia
- No se deben utilizar ciclos dentro de ciclos. Use funciones para hacerse la vida más fácil.
- Use la arquitectura vista en clase: **DEBE** crear un “Sistema”.
- Recuerde que toda la lectura de datos y salidas a la pantalla se hacen desde la App.
- El código fuente debe comprimirlo en un solo archivo .zip y subirlo a Campus Virtual.
- Archivos .txt de ejemplo se encuentran en CampusVirtual
- **Modele los ingresos con o sin administración con el patrón Visitor.**

Ejemplo de ejecución:

Ingresos sin Administración

Casa Independiente: \$47874736
Casa en Condominio: \$62608024
Casa de Uso Comercial: \$64939882
Departamento: \$10971252
Espacio Comercial: \$337882287

Ingresos con Administración

Casa Independiente: \$13300525
Casa en Condominio: \$15072210
Casa de Uso Comercial: \$18554252
Departamento: \$3657084
Espacio Comercial: \$84470572

Problema 2. (50 puntos)

El hotel EIC necesita ayuda para mejorar sus ganancias, por lo que requiere de usted un programa que procese un archivo que ha sido creado para tal efecto y que contiene la configuración del hotel y las personas que han llegado a alojar.

Este archivo (llamado **config.txt**) tiene la siguiente estructura:

config.txt
5
C,4
S
S
C,3
C,1
PREMIUM:ximena,NORMAL:pedro,NORMAL:juan,NORMAL:paula,NORMAL:diego,PREMIUM:katy,PREMIUM:klaus
20000

La primera línea contiene un número (**N**) que indica la cantidad de habitaciones del hotel. Después vienen N líneas, cada una indicando si la habitación indicada es “compartida” (**C**) o “individual” (**S**). Si es compartida, se indica la cantidad máxima de personas que pueden alojar en dicha habitación. Si la habitación es “**S**” entonces solo puede alojar una persona.

Después viene una sola línea con los pasajeros que llegaron al hotel. Cada pasajero viene indicado por su categoría (**PREMIUM** o **NORMAL**) y su nombre. Los pasajeros se procesan en el orden en que vienen en la línea.

Finalmente, la última línea indica el precio por persona por alojar en el hotel.

Cuando una persona llega al hotel, el sistema debe tratar de alojar a la persona en alguna habitación, en el orden en que las habitaciones aparecen en el archivo, pero se deben tener en consideración las siguientes reglas:

- Las personas **PREMIUM** pueden alojar en cualquier habitación, pero siempre que no la comparta con nadie. Vea el ejemplo más abajo.
- Las personas **NORMAL** puede alojar en cualquier habitación siempre que exista espacio.
- Si a una persona no se le puede encontrar habitación para alojar, queda en la “lista de espera”.

La aplicación debe entregar por pantalla la configuración de cada habitación del hotel, y la “ganancia”, que corresponde al valor del “precio por persona” multiplicado por la cantidad real de personas alojadas en el hotel.

Debe entregar:

- Modelo de Dominio (10%)
- Diagrama de Clases (15%)
- Código Java, incluyendo contratos en **javadoc** (75%)

Consideraciones:

- Debe usar orientación al objeto. Debe usar herencia
- No se deben utilizar ciclos dentro de ciclos. Use funciones para hacerse la vida más fácil.
- En el diagrama de clases debe especificar **TOD**O. No es buena práctica especificar todo público, ni tampoco crear getters o setters para todos los atributos. No se aceptarán “gets” y “sets”.
- Use la arquitectura vista en clase: **DEBE** crear un “Sistema”.
- Recuerde que toda la lectura de datos y salidas a la pantalla se hacen desde la App.
- Tanto el modelo del dominio, como el diagrama de clases debe escribirlos en papel y entregarlos junto a la prueba.
- El código fuente debe comprimirlo en un solo archivo .zip y subirlo a Campus Virtual.
- Hojas sin nombre no se revisarán.
- Archivos .txt de ejemplo se encuentran en CampusVirtual
- **Modele el sistema como un Singleton**
- **Modele la creación de los diferentes tipos de pasajeros y habitaciones como Factory**

Ejemplo de ejecución:
Habitación Compartida (4): PREMIUM ximena
Habitación Simple NORMAL pedro
Habitación Simple NORMAL juan
Habitación Compartida (3): NORMAL paula NORMAL diego
Habitación Compartida (1): PREMIUM katy
Lista de espera: PREMIUM klaus
Ganancia: 120000