

## **Primer Parcial**

### **Objetivo**

Evaluar al estudiante en la parte práctica de la unidad nro. 4, Paradigma Orientado a Objetos, de los siguientes temas: Clases, objetos, paso de mensajes, composición, herencia, polimorfismo y colecciones en Smalltalk.

### **Condiciones de trabajo:**

- El alumno deberá desarrollar un programa, utilizando el correspondiente entorno.
- El código fuente del alumno deberá quedar guardado en la unidad D: nominado con su legajo y nombre, de la siguiente forma: Legajo\_ApellidoNombre.st.
- **Es responsabilidad de cada alumno ir guardando periódicamente su archivo, como así también del contenido de los mismos, teniendo la precaución de guardarlo en el disco D: para su posterior backup.**
- En caso de que máquina no funcione correctamente durante el transcurso de la evaluación, debe notificar de esta situación a cualquier docente de la mesa examinadora.
- En ningún caso debe reiniciar la máquina, ya que perderá la totalidad del examen.
- **Es responsabilidad de cada estudiante el contenido que quede guardado al momento de finalizar la evaluación, en todos sus archivos generados durante el examen.**
- El **tiempo previsto** para la realización de este examen es de **1:30 hs.**

## Tema 2

### Enunciado:

Una institución pedagógica nos ha solicitado desarrollar un sistema a través del cual se puedan registrar los exámenes evaluados a sus alumnos de forma de obtener resultados y sacar conclusiones con respecto a sus instrumentos de evaluación.

#### Importante

La institución nos informa que los exámenes además de los datos del alumno contienen un set de preguntas. Las preguntas pueden ser de múltiple opción (donde una sola opción es la correcta) o bien, una pregunta cuya respuesta se espera sea redactada por el alumno.

Para ello se han identificado las siguientes entidades:

**Pregunta**, tiene como atributos:

- número de pregunta
- texto: descripción de la pregunta en sí, por ejemplo, “qué es un objeto?”
- puntaje: puntaje asignado a la pregunta dentro del examen (por ejemplo, dentro del examen la pregunta tiene asignado 10 puntos).

Con los siguientes métodos:

1. Constructor
2. Métodos de acceso y modificación
3. Mostrar datos: que retorna los valores de los atributos de la clase concatenados en un objeto string.

**Redactada**, clase hija de Pregunta, que representa una pregunta a desarrollar, es decir, cuya respuesta debe ser redactada por un alumno. Esta clase además de los atributos heredados tiene como atributos propios:

- respuestaAlumno: respuesta dada por el alumno (de texto libre) y,
- evaluaciónDocente: evaluación dada por el docente que puede ser [Mal/Regular/Bien/Muy Bien/Excelente].

Con los siguientes métodos:

1. Constructor
2. Métodos de acceso y modificación
3. Mostrar datos: que retorna los datos de los atributos heredados y propios concatenados en un objeto string. Además, se debe agregar el puntaje obtenido por el alumno (usar el método puntajeAlumno).
4. puntajeAlumno: el sistema determina cual es el puntaje obtenido por el alumno en la pregunta. Para ello se tiene en cuenta el siguiente criterio: Si la evaluación dada por el docente es:
  - “Mal”: retorna cero.
  - “Regular”: retorna el 45 % del puntaje total que tiene asignado la pregunta.
  - “Bueno”: retorna el 55 % del puntaje total que tiene asignado la pregunta.
  - “Muy bueno”: retorna el 80 % del puntaje total que tiene asignado la pregunta.
  - “Excelente”: retorna el puntaje total que tiene asignada la pregunta en el examen.

**MultipleChoice**, clase hija de Pregunta, que representa una pregunta con cuatro opciones de las cuales solo una opción es la correcta. Esta clase además de los atributos heredados tiene como atributos propios:

- una colección con las opciones posibles (Instancias de la clase Opción)
- opciónElegidaAlumno: número de la opción seleccionada por el alumno
- opciónCorrecta: número de la opción correcta.

Con los siguientes métodos:

1. Constructor
2. Métodos de acceso y modificación
3. Mostrar datos: que retorna los datos de los atributos heredados y propios concatenados en un objeto string. Además, se debe agregar el puntaje obtenido por el alumno (usar el método puntajeAlumno).
4. puntajeAlumno: en este caso se debe tener en cuenta que si el alumno responde mal se le resta el puntaje que tiene asignado la pregunta. Por lo tanto:
  - Si la opción elegida por el alumno es correcta este método retorna el puntaje total asignado a la pregunta
  - Si la opción elegida es "incorrecta" retorna el puntaje que tiene asignado la pregunta en valor negativo
  - Si la pregunta no tiene opción elegida por el alumno (es decir, que el atributo opción elegida por el alumno es cero), debe retornar cero.

**Opción**, representa una opción de una pregunta multiplechoice que posee como atributos: número de opción y descripción de la opción. Con los siguientes métodos:

1. Constructor
2. Métodos de acceso y modificación
3. Mostrar datos: que retorna los datos de los atributos concatenados en un objeto string.

**Alumno**, representa al alumno que rinde el examen. Los atributos que se registran del alumno son: legajo (entero) y nombre. Con los siguientes métodos:

1. Constructor
2. Métodos de acceso y modificación
3. Mostrar datos: que retorna los datos de los atributos concatenados en un objeto string.

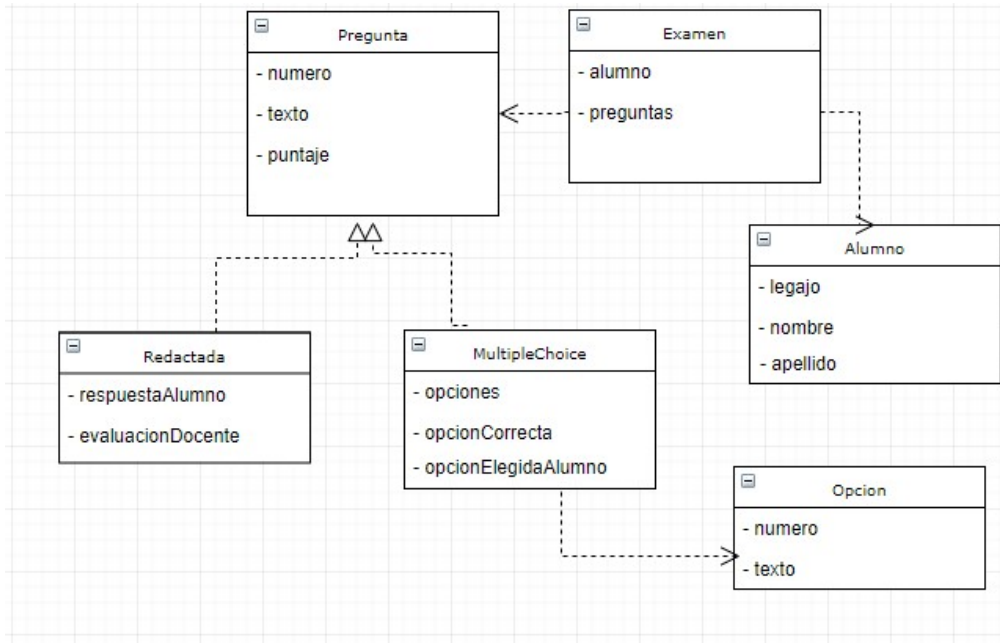
**Examen**, representa el examen rendido por un alumno, que posee como atributos:

- un alumno (instancia de la clase Alumno) y,
- una colección con las preguntas que se evalúan.

Esta clase posee los siguientes

1. porcentajeFinal: método que retorna el porcentaje total obtenido en el examen. Para ello, la suma del puntaje asignado a todas las preguntas constituye el 100%, y se debe determinar el porcentaje que representa el puntaje total obtenido por el alumno.
2. preguntasPorPuntaje: método que retorna una colección con las preguntas con un puntaje obtenido por el alumno comprendido en el intervalo desde – hasta (donde desde y hasta son valores pasados por parámetro).

3. mostrarPreguntasChoice: que muestra los datos completos de las preguntas multiplechoice que no estén aprobadas (la opción elegida por el alumno no es igual a la opción indicada como correcta).



### Requerimientos del parcial:

Con lo expuesto anteriormente, usted deberá implementar:

1. Las clases Alumno y Opción, con los métodos de inicialización, acceso y modificación y, asString.
2. La clase base Pregunta y sus clases hijas, con los métodos de inicialización, acceso y modificación y, asString.
3. Los métodos que considere necesarios para retornar el puntaje obtenido por el alumno en la pregunta.
4. La clase Examen con los métodos de inicialización, acceso y modificación y, el método addPregunta que permite agregar una pregunta a la colección.
5. El método 1 pedido en Examen.
6. El método 2 pedido en Examen.
7. El método 3 pedido en Examen.

## 1. Tabla de valoración de los ítems evaluados

Ítems a Evaluar	Pje	Observaciones	Obtenido
Implementación de la jerarquía de clases, Alumno y Opción, con métodos comunes.	25		
Definición de métodos polimórficos en la jerarquía	15		
Implementación de métodos comunes en la clase Examen	10		
Implementación del método 1 en Examen	15		
Implementación del método 2 en Examen	15		
Implementación del método 3 en Examen	20		
Total	100		