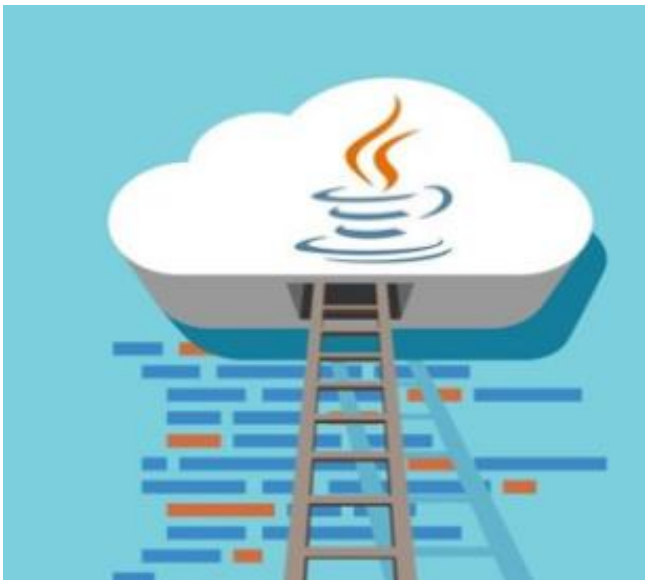




UNIVERSIDAD DE
LA SERENA



Informe Proyecto Programación Orientada a Objetos.



Matías Jesús Egaña Alfaro.

Dr. Eric Roberto Jeltsch Figueroa.

Ingeniería en Computación.

Universidad de La Serena.



UNIVERSIDAD DE
LA SERENA



Introducción

¿Qué es la Programación Orientada a Objetos?

Es un paradigma de la programación que utiliza objetos para emular entidades de la vida real.

Este curso se rige principalmente por el lenguaje de programación Java y este proyecto tiene como objetivo aplicar los conceptos de la POO en su más amplia extensión.

Instrucciones iniciales

Crear un sistema capaz de administrar un examen tipo cuestionario, basado en 3 tipos de preguntas. Se fijan entre ellas, preguntas del tipo

- a) Verdadero/falso,
- b) Selección múltiple,
- c) Respuesta cortas.

Cada pregunta debe tener un puntaje o peso, el que representará el resultado del examen.

En concreto, su propuesta deberá ser capaz de crear un examen con no más de 10 preguntas, las cuales son matizadas según lo antes señaladas, con un total de 3 preguntas de cada ítem, por ejemplo.

Luego deberá administrar (el examen) en el sentido de dar respuestas a las distintas preguntas que se plantearon. Los tipos de preguntas se deben circunscribir a una temática simple, para finalmente reportar los resultados en porcentaje.

Extra: Podrá llevar un registro del rendimiento de los usuarios al sistema, por ejemplo, con uso de archivos.

Preparación

Lo primero fue ver las instrucciones y seguir las propuestas dadas por la pauta.

En esta estaban los requisitos de crear **la clase abstracta Pregunta** y a partir de esta, crear **extensiones** de la clase. Estas son: **TF_Pregunta**, **Resp_Cortas_Pregunta**, **Selec_Mul_Pregunta**.

También una clase que entrega una colección de preguntas para el examen este se denomina **Exam.java**.



UNIVERSIDAD DE
LA SERENA

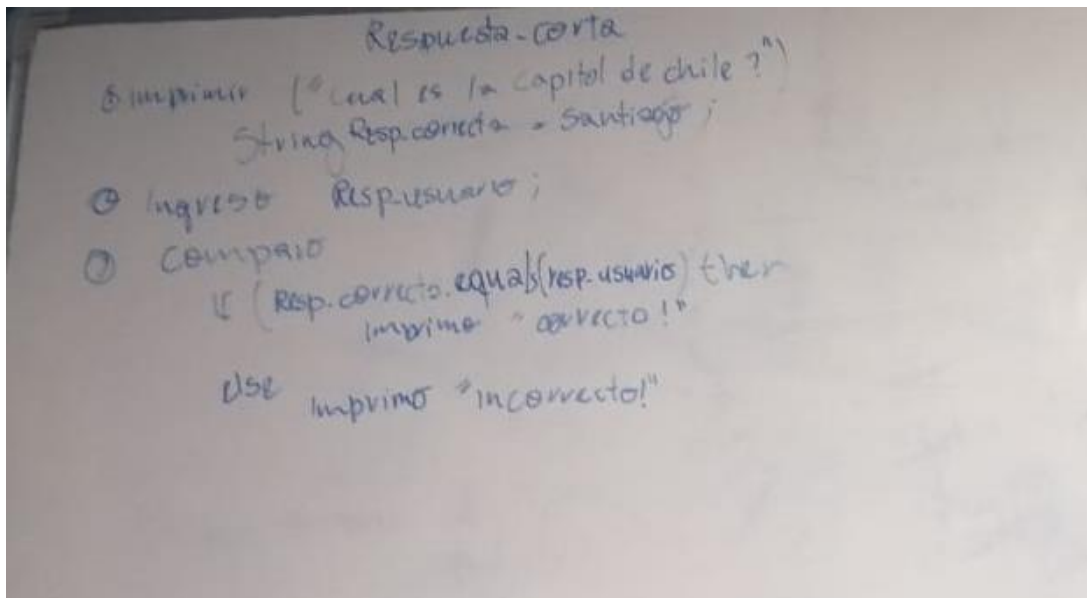


Y para ejecutar el programa su respectiva clase testeadora **ExamDemo.java**

Comienzo

Al principal el primer problema fue el ¿cómo comenzar?, ya que no dominaba el lenguaje ni ordenar mis ideas, ya que hago demasiados borradores y en esa voy perdiendo la idea, por eso implemente en colocar una pizarra y en esta iba añadiendo mis ideas en pseudocódigo.

Procedo a adjuntar mis pseudocódigos antes de comenzar con la codificación:



Pseudocódigo de la clase **Resp_Cortas_Pregunta;**



UNIVERSIDAD DE
LA SERENA



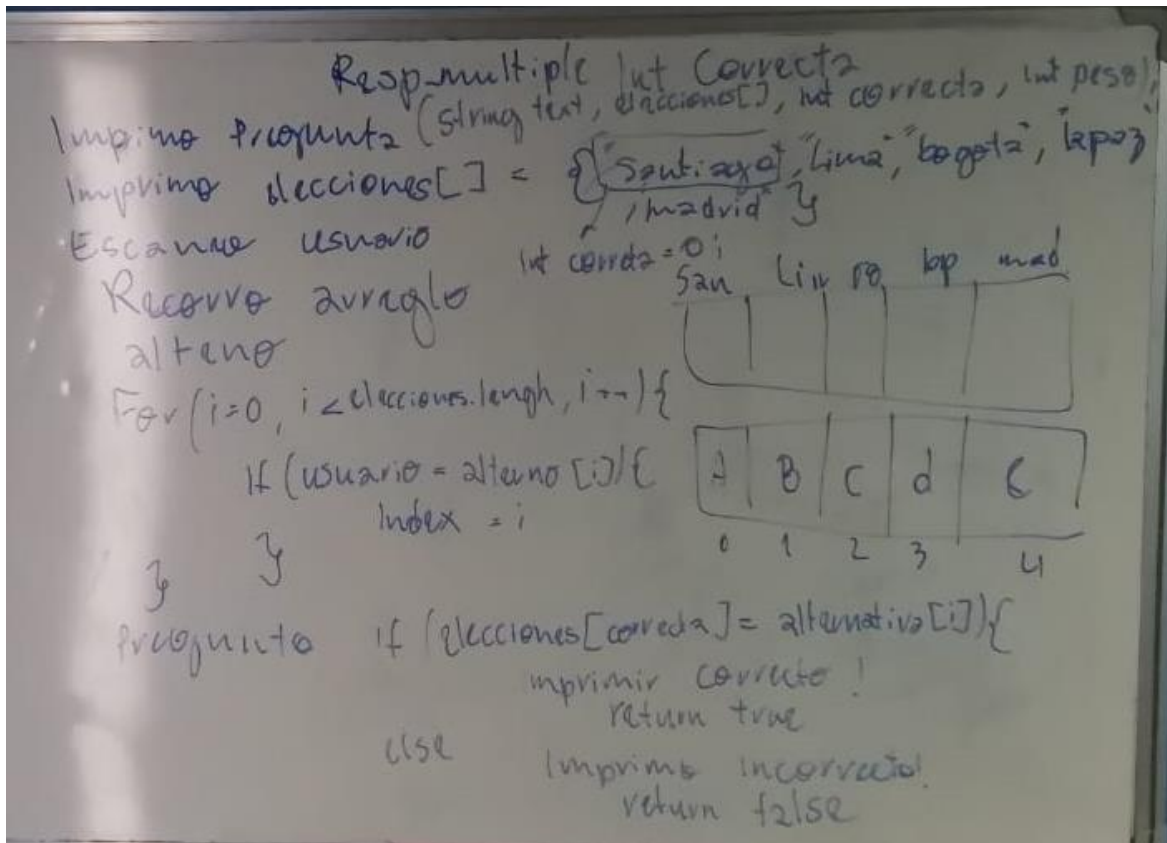
TF_Pregunta

```
Imprimir "Santiago es la capital de Chile";  
ingrese T o F          boolean respuesta = true  
Escriba respuestaUsuario;  
if (respuestaUsuario.equalsIgnoreCase('T')) {  
    if (resp.correcta == true) {  
        Imprime Correcto!  
        return True  
    }  
    else Imprime Incorrecto!  
        return false  
} else if (respuestaUsuario.equalsIgnoreCase('F')) {  
    if (resp.correcta == false) {  
        Imprime Correcto!  
        return true;  
    }  
    else { Imprime Incorrecto;  
        return false;  
    }  
}
```

Pseudocódigo de la clase **TF_Pregunta**;



UNIVERSIDAD DE
LA SERENA



Pseudocódigo de la clase **Selec_Mul_Pregunta**

Posteriormente, divido cada clase en proyecto distintos por separados la cual denomino "**ExamDemo Ver**", para empresa a codificar por partes, ya que me resultaría más fácil dividir cada clase, pero con la clase **Pregunta.java** en común y finalmente procedo a unirlos todos los proyectos ya testeados y asegurando que ejecutan cada uno por separado, en el proyecto final "**ExamDemo vFinal**".

- ExamDemo v2
- ExamDemo V3
- ExamDemo v4
- ExamDemo ver 0
- ExamDemo ver 1
- ExamDemo vFinal



UNIVERSIDAD DE
LA SERENA



Codificación

Entre esto creo la clase abstracta **Pregunta**, siguiendo los requisitos iniciales, estos son:

Una variable **peso** para el "peso" de la pregunta, además de métodos **public** **getPeso ()** y **setPeso ()** para esta variable.

Una variable **text** para el texto de la pregunta, además de un método **public** **getText ()** para esta variable.

Un método abstract **buscar ()**, que pedirá la pregunta, lee la respuesta del usuario y return un resultado boolean indicando si el usuario la respondió correctamente. También deberá imprimir un mensaje al usuario, llamando la atención de que el resultado a la pregunta fue correcto o no.

```
Pregunta.java
1 package ExamDemo;
2
3 public abstract class Pregunta {
4     /* Variable peso, privado para usarlo solo en esta clase y
5      para usarlos fuera de la clase se usa get y set
6
7      */
8     private String text;
9
10    /* Una variable text para el texto de la pregunta */
11
12    public Pregunta(String text) {
13        this.text = text;
14    } // constructor Pregunta
15
16    private int peso;
17
18    public int getPeso() {
19        return this.peso;
20    }
21
22    public void setPeso(int pes) {
23        this.peso = pes;
24    }
25
26    public String getText() {
27        return text;
28    }
29
30    public abstract boolean buscar();
31
32
33    /*
34     Pedirá la pregunta, lee la respuesta del usuario
35     y return un resultado boolean indicando si el usuario la respondió correctamente.
36     */
37
38 }
39
```



UNIVERSIDAD DE
LA SERENA



```
public class Resp_Cortas extends Pregunta {
    private String resp_correcto;

    public Resp_Cortas(String text,String correcto,int peso){
        super(text);
        this.resp_correcto = correcto;
        super.setPeso(peso);
    }

    public boolean buscar() {
        System.out.println(getText() + " Ingrese la respuesta que crea usted correcta ");
        String resp_usuario = SavitchIn.readLine().toLowerCase();
        if(resp_correcto.equalsIgnoreCase(resp_usuario)) {
            System.out.println("La Respuesta es Correcta!");
            return true;
        }
        else {
            System.out.println("Respuesta Incorrecta!");
            return false;
        }
    }
}
```



UNIVERSIDAD DE
LA SERENA



```
public class TF_Pregunta extends Pregunta {
    private boolean Legal;
    public TF_Pregunta(String text , boolean respuesta, int peso){
        super(text);
        this.Legal = respuesta;
        super.setPeso(peso);
    }
    public boolean buscar() {
        System.out.print(getText() + " (Ingrese T o F): ");
        // obtener respuesta del usuario y traslada a to lower case
        String Usuario = SavitchIn.readLine().toLowerCase();
        if (Usuario.equalsIgnoreCase("t")) {
            if (Legal==true) {
                System.out.println(" La Respuesta es Correcta! ");
                return true;
            }
            else {
                System.out.println( " Respuesta Incorrecta! ");
                return false;
            }
        }
        else
            if (Usuario.equalsIgnoreCase("f")) {
                if (Legal==false) {
                    System.out.println(" La Respuesta es Correcta! ");
                    return true;
                }
                else {
                    System.out.println(" Respuesta Incorrecta! ");
                    return false;
                }
            }
        else { // respuesta no valida true/false
            System.out.println(" Respuesta Invalida, Debe ingresar 'T' o 'F' e Intentelo nuevamente. ");
            return buscar(); // otra vez
        }
    }
}
```




```
public class Selec_Mul_Pregunta extends Pregunta{

    private String[] array_elecciones;
    private int correcta;
    String[] alternativas = {"a","b","c","d","e"};
    public Selec_Mul_Pregunta(String text,String[] array,int respuesta_correcta,int peso){
        super(text);
        this.array_elecciones = array;
        this.correcta = respuesta_correcta;
        this.setPeso(peso);
    }

    public boolean buscar() {
        int index = 0;
        int i=0;
        System.out.println(getText());
        for(String j:array_elecciones) { //imprimo alternativas
            System.out.println(j);
        }
        System.out.println("Ingrese la alternativa que crea usted correcta");
        String respuestaUsuario = SavitchIn.readLine().toLowerCase();
        for(i=0 ; i<array_elecciones.length ; i++) {
            if(respuestaUsuario.equalsIgnoreCase(alternativas[i])){
                index = i;
            }
        }

        if(array_elecciones[correcta].equalsIgnoreCase(array_elecciones[index])) {
            System.out.println("Correcta!");
            return true;
        }
        else {
            System.out.println("Incorrecta!");
            return false;
        }
    }
}
```



UNIVERSIDAD DE
LA SERENA



Ejecuciones

```
1 package ExamDemo;
2
3 public class ExamDemo {
4     public static void main(String[] args) {
5         TF_Pregunta a = new TF_Pregunta("¿Santiago es la capital de chile?", true, 1);
6         boolean f = a.buscar();
7
8     }
9 }
10
11
12
```

Console

<terminated> ExamDemo [Java Application] C:\Users\mati_\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_14

¿Santiago es la capital de chile? (Ingrese T o F): t

La Respuesta es Correcta!

```
2
3 public class ExamDemo {
4     public static void main(String[] args) {
5         Resp_Cortas b = new Resp_Cortas("¿ Cuánto es 64+4 ?", "68", 1);
6         boolean g = b.buscar();
7
8     }
9 }
10 }
11
```

Console

<terminated> ExamDemo [Java Application] C:\Users\mati_\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot

¿ Cuánto es 64+4 ? Ingrese la respuesta que crea usted correcta

68

La Respuesta es Correcta!



UNIVERSIDAD DE
LA SERENA



```
1 package ExamDemo;
2
3 public class ExamDemo {
4     public static void main(String[] args) {
5         String[] santiagoelec = {"a. Barcelona", "b. Madrid", "c. Santiago", "d. Lima", "e. Bogota"};
6         Selec_Mul_Pregunta c = new Selec_Mul_Pregunta("Cuál es la capital de Region Metropolitana", santiagoelec, 2, 1);
7         boolean h = c.buscar();
8     }
9 }
10
11
```

Console

<terminated> ExamDemo [Java Application] C:\Users\mati_\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe

Cuál es la capital de Region Metropolitana

- a. Barcelona
- b. Madrid
- c. Santiago
- d. Lima
- e. Bogota

Ingrese la alternativa que crea usted correcta

c

Correcta!

En este punto el programa funciona .Procedo a unir las partes, en el proyecto final y a esta le creo la clase ExamDemo para poder ir coleccionado las preguntas y que nos entregue el puntaje. Para esta clase debo decir que fue la que me costo y la que tardo más dedicación y empeño, ya que no sabia como modelarla , pero gracias a investigaciones y ayuda de ciertos compañeros se pudo llevar a cabo exitosamente esta clase.



UNIVERSIDAD DE
LA SERENA



```
public class Exam {
    /*Es una clase que entrega una colección de preguntas para un examen dado. El examen parte vacío (no existen preguntas),
    y proporciona un método para agregar preguntas a un examen. Esta clase deberá contener:
    Un array de preguntas.
    Estudie la forma de limitar el nº límite de preguntas, por ahora cree un array de tamaño 10.
    Un contador para el nº de preguntas en el array.
    Un método agregaPregunta(), el que toma una pregunta (de cualquier tipo) como argumento y la agrega al examen.

    Un método darExam() el cual administra el examen al usuario, con el fin de garantizar el nº de respuestas,
    con su porcentaje y el redondeo apropiado, como resultado.
    */

    private Object[] Array_Preguntas = new Object[10];
    private int PuntajeUsuario=0;
    private int PuntajeTotal=0;
    private int n_Pregunta=0;

    public void agregaPregunta(Object pregunta) {
        if(n_Pregunta<10) {
            Array_Preguntas[n_Pregunta] = pregunta;
            n_Pregunta++;
        }
        else {
            System.out.println("Error limite excedido");
        }
    }

    public int DarExam() {
        System.out.println("*****Examen*****");
        boolean correccion;
        int Resultado = 0;
        int i = 0;
        for(i=0;i<n_Pregunta;i++) {
            if(Array_Preguntas[i].getClass() == Resp_Cortas.class) {
                Resp_Cortas respuestacorta = (Resp_Cortas) Array_Preguntas[i] ;
                // System.out.println(""+respuestacorta.getText()+"(respuestacorta)");
                correccion = respuestacorta.buscar();
                if(correccion) {
                    PuntajeUsuario = PuntajeUsuario + respuestacorta.getPeso();
                }
                PuntajeTotal = PuntajeTotal + respuestacorta.getPeso();
            }
            if(Array_Preguntas[i].getClass() == TF_Pregunta.class) {
                if(Array_Preguntas[i].getClass() == TF_Pregunta.class) {
                    TF_Pregunta tfpregunta = (TF_Pregunta) Array_Preguntas[i] ; //creo objeto y la inicializo en el arreglo;
                    // System.out.println(""+tfpregunta.getText()+"(tfpregunta)");
                    correccion = tfpregunta.buscar();
                    if(correccion) {
                        PuntajeUsuario = PuntajeUsuario + tfpregunta.getPeso();
                    }
                    PuntajeTotal = PuntajeTotal +tfpregunta.getPeso();
                }
            }
            if(Array_Preguntas[i].getClass() == Selec_Mul_Pregunta.class) {
                Selec_Mul_Pregunta selecmulpregunta = (Selec_Mul_Pregunta) Array_Preguntas[i] ;
                // System.out.println(""+selecmulpregunta.getText()+"(selecmulpregunta)");
                correccion = selecmulpregunta.buscar();
                if(correccion) {
                    PuntajeUsuario = PuntajeUsuario + selecmulpregunta.getPeso();
                }
                PuntajeTotal = PuntajeTotal +selecmulpregunta.getPeso();
            }
        }
        double Porcentaje = (PuntajeUsuario*100)/PuntajeTotal;
        Resultado = (int)Porcentaje;
        System.out.println("El Puntaje Obtenido es: " +PuntajeUsuario);
        System.out.println("El Puntaje Total es: " +PuntajeTotal);
        return Resultado;
    }
}
```

Clase Exam.Java



UNIVERSIDAD DE
LA SERENA



Ahora con esto terminado lo unico que queda es la clase testeadora **ExamDemo.Java**.

```
public class ExamDemo {
    public static void main(String[] args) {
        Exam miExam = new Exam();
        miExam.agregaPregunta(new TF_Pregunta("¿La capital de Chile es Santiago?", true, 1));
        String santiagoElec[] = {"a. Santiago", "b. Valparaiso", "c. Copiapo", "d. Magallanes", "e. Temuco"};
        miExam.agregaPregunta(new Selec_Mul_Pregunta("¿Cuál es la capital de Region Metropolitana?", santiagoElec, 0, 1));
        miExam.agregaPregunta(new Resp_Cortas("¿Cuál es la capital de IV Region?", "La Serena", 1));
        miExam.agregaPregunta(new TF_Pregunta("¿La capital de Estados Unidos es New York?", false, 1));
        miExam.agregaPregunta(new Resp_Cortas("¿ Cuánto es 64+4 ?", "68", 1));
        miExam.agregaPregunta(new Resp_Cortas("¿Cuál es la capital de Alemania?", "Berlin", 2));
        miExam.agregaPregunta(new TF_Pregunta("¿La capital de Rusia es Stalingrado?", false, 1));
        String PElec[] = {"a. Brasilia", "b. Rio de Janeiro", "c. Sao Paulo", "d. Lima"};
        miExam.agregaPregunta(new Selec_Mul_Pregunta("¿Cuál es la capital de Brasil?", PElec, 0,1));

        int puntaje = miExam.DarExam();
        System.out.println("El Porcentaje de respuestas correctas es: " + puntaje + "%");
    }
}
```

Clase Testeadora ExamDemo.Java.

Testeo en Otras Persona.

Lo teste en otras personas y este fue el resultado de algunas personas de mi entorno.

El Puntaje Obtenido es: 8

El Puntaje Total es: 9

El Porcentaje de respuestas correctas es: 88%

Mi hermana obtuvo un 88%

El Puntaje Obtenido es: 5

El Puntaje Total es: 9

El Porcentaje de respuestas correctas es: 55%

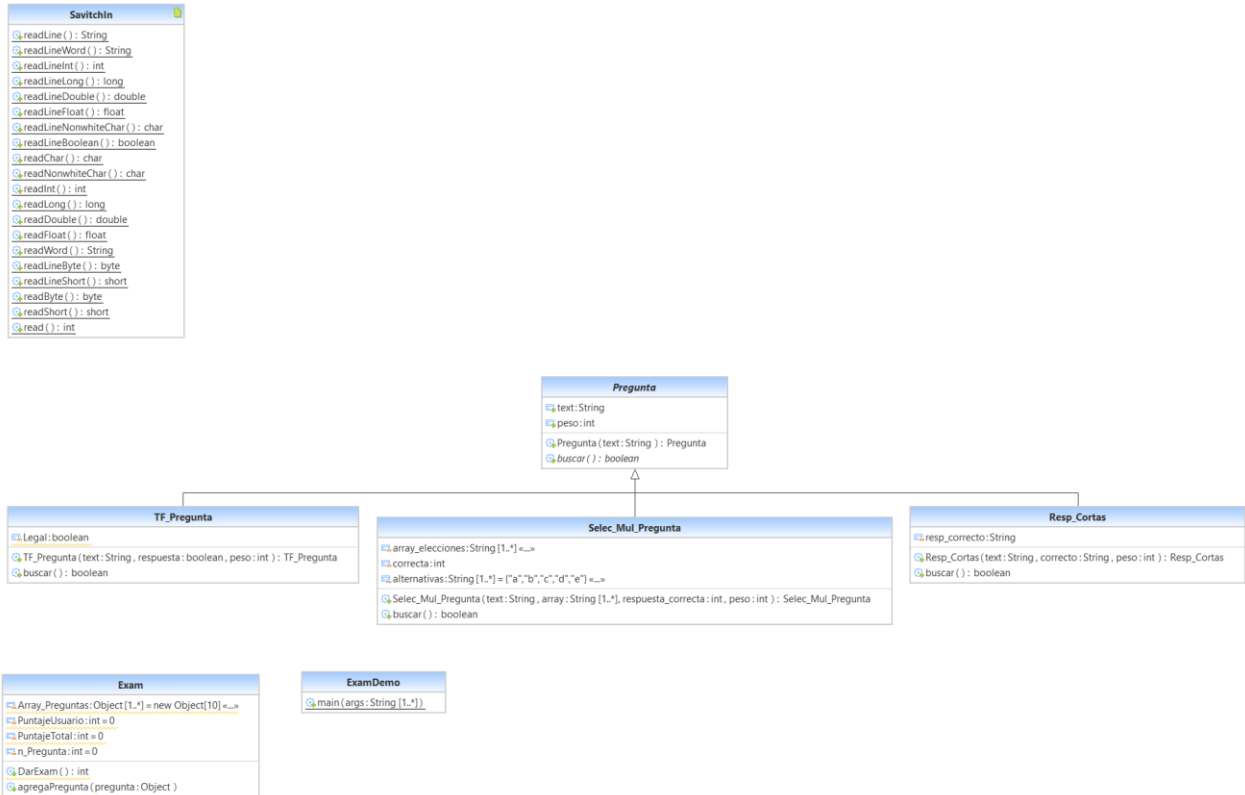
Y mi padre Obtuvo un 55%



UNIVERSIDAD DE
LA SERENA



Diagrama UML



Conclusión

Este proyecto me pareció bastante interesante y ayudó a reforzar mis habilidades de programación, si bien puede que hayan algunos errores no encontrados, pero al menos es un gran avance para mí, el empezar un nuevo lenguaje de programación y lograr crear algo por mí.