

**Pensamiento Computacional Orientado a Objetos**

**Nombre del profesor:**

*Sasha Valdés Jiménez*

**Actividad Complementaria - Diseño de la Situación Problema**

23 de noviembre del 2023

Grupo 301

**Equipo 2 | Integrantes:**

Amparo Alcaraz Tonella	A01253885
Diana Escalante García	A01253468
Kimberly Gabriela Marquez Preciado	A01255262
Libia Zulema Flores Valenzuela	A01253425

## **PARTE I. MARCO TEÓRICO**

### **1.1 Situación problemática**

La educación en México es, sin duda, un problema relevante que cuenta con muchos desafíos y retos por delante, la deserción escolar ha aumentado bastante, especialmente desde la pandemia, ya que se estima que 5.2 millones de alumnos dejaron los estudios (AP, 2021). Sin embargo, esta no es la única razón. Una gran cantidad de niños, adolescentes y adultos se ven obligados a abandonar sus estudios por distintos motivos, ya sean económicos, de vivienda, discapacidad, entre otros, lo que vuelve al sistema no justo y desigual. Esta desigualdad en la educación “conduce a que la educación no actúe como mecanismo de igualación social” (INEE, 2018).

En México, existe una gran brecha de desigualdad en el área educativa, en comunidades rurales 6 de cada 15 jóvenes no cuentan con escuelas cercanas a ellos (García, 2018). Incluso, el 13.2% de niños y jóvenes en extrema pobreza no asisten a la educación obligatoria (García, 2018). En muchas ocasiones, no se lleva un control de cómo va la educación en zonas aisladas. Por lo que estas quedan fuera y no se puede dar mucho apoyo, debido a que no se tiene información que sirva como indicador para medir el nivel de educación de la zona. Con el propósito de tener estadísticas que respalden la gravedad del problema.

Tomando en cuenta esta problemática, se busca crear un sistema que sea capaz de obtener el porcentaje de las personas graduadas de la universidad dada una muestra de la población. Esto con el fin de obtener una idea del porcentaje de personas con estudios superiores en una zona específica en el país. De esta manera, se puede obtener una idea de a qué nivel de educación logran llegar las personas, y aproximar cuántas personas cuentan con sus estudios superiores terminados en la población. Basado en el porcentaje de personas con sus estudios superiores terminados, se crea una alerta si este es muy bajo, con el propósito de poder tomar acción para resolver esto y resolver problemas con la educación en esa zona.

## **1.1 Propuesta de Solución**

Definitivamente, uno de los principales problemas con la educación es México, es su sistema. Igualmente, como en ocasiones no se cuenta con información de la educación en algunas zonas del país, especialmente las aisladas. Nuestro propósito es lograr recabar información y contar con datos sólidos que en un futuro contribuyan a nuevas propuestas e iniciativas para mejorar la educación en México. Nuestra propuesta para resolver esta situación problema, es crear un programa que funcione como un sistema para medir el porcentaje de las personas que han concluido sus estudios universitarios de una cierta edad. Es decir, se tomará un grupo de personas de una misma edad, habitantes de una zona, y se sacará el porcentaje de personas graduadas del grupo. De igual manera, de ese porcentaje de personas graduadas, cuál es el porcentaje de mujeres y hombres. Esto con el fin de presentar que tan alto o bajo está el porcentaje de personas con estudios de educación superior terminada en una zona. Al igual que la diferencia de porcentajes que existe entre hombres y mujeres. La información se extraerá de un fichero y al final del programa se proporcionará la información que fue recolectada. Se usará el lenguaje de programación de C++ para desarrollar este programa. Se contará con tres clases, una siendo la zona, otra la población que lleva a un arreglo de evaluados.

## 1.2 Funcionalidades

Especifique no más de una funcionalidad por integrante. Explique el objetivo de la funcionalidad y su nivel de contribución. Puede incluir gráficas de resultados o tablas.

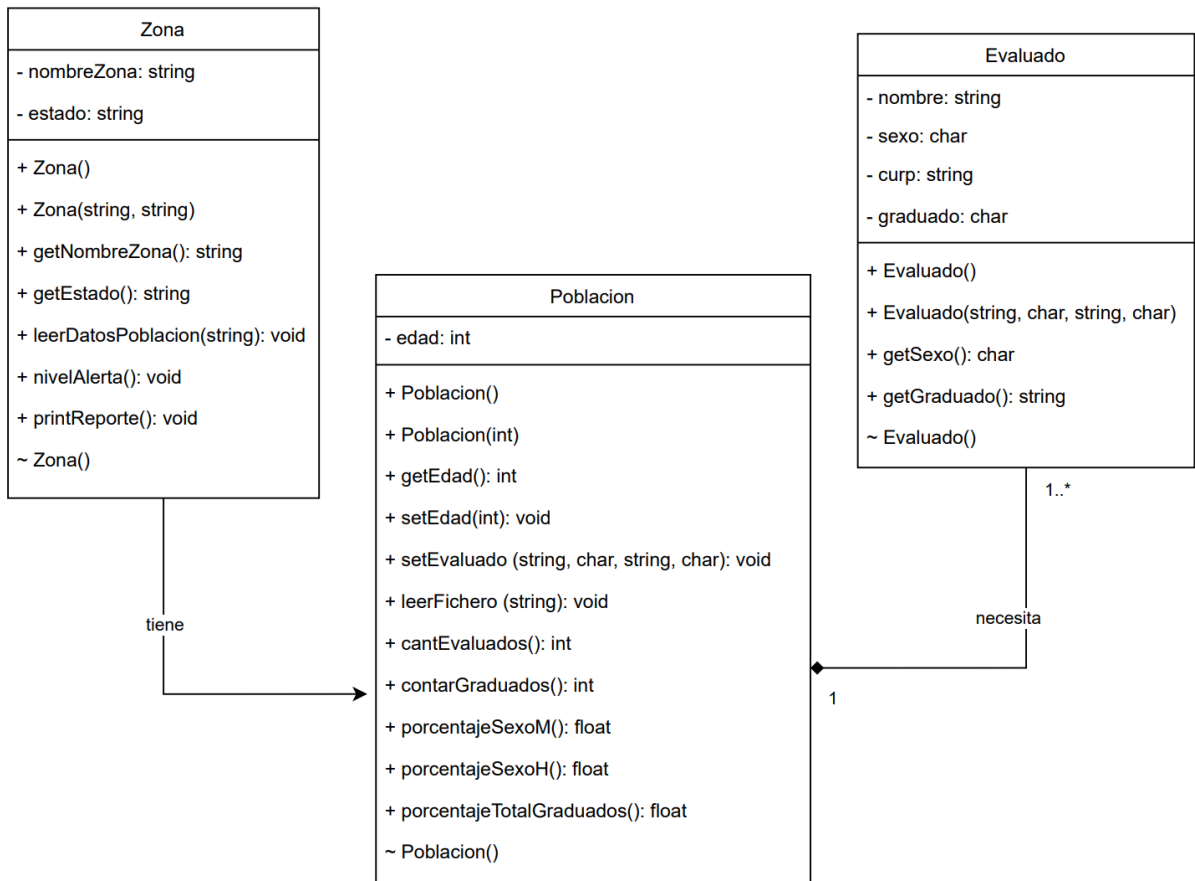
Funcionalidad	Explicación
+ leerFichero(string) : void	Su función es leer el fichero que contiene los datos del total de las personas evaluadas del mismo grupo de edad (nombre, sexo, CURP y si se graduaron de la carrera o no)
+ contarGraduados() : int	Con ayuda de esta funcionalidad se busca contar la cantidad de personas graduadas dentro del arreglo de evaluados. Esta será de gran utilidad para el cálculo del porcentaje de graduados de cierta edad.
+ porcentajeSexoM() : float	Su propósito es conseguir el porcentaje de evaluados de sexo femenino que se han graduado de los estudios superiores, específicamente la universidad. Este formará parte del resultado y reporte que se proporcionará al usuario.
+ porcentajeSexoH() : float	De manera similar a la funcionalidad anterior, se busca obtener el porcentaje de evaluados de sexo masculino que se han graduado de sus estudios universitarios. Lo que se le proporcionará al usuario después.
+ cantEvaluados(): int	Retorna el tamaño del arreglo de tipo Evaluado que contiene toda la información de

	los evaluados.
+ porcentajeTotalGraduados() : float	Divide el número de personas graduadas de la universidad por el número total de las personas del mismo grupo de edad evaluado, y se multiplica por 100. Regresa el porcentaje de personas que han finalizado la universidad.
+ leerDatosPoblacion(string): void	Esta función accede y lee los valores del fichero con la información.
+ nivelAlerta() : string	Devuelve el nivel de alerta de la zona respecto a los resultados obtenidos, por medio de condiciones: <ul style="list-style-type: none"> <li>- Si el porcentaje es menor al 20%, el nivel de alerta es grave</li> <li>- Si el porcentaje está entre el 20% y el 50%, el nivel de alerta es alto</li> <li>- Si el porcentaje está entre el 50% y el 80%, el nivel de alerta es bajo.</li> <li>- Si el porcentaje es mayor al 80%, el nivel de alerta es muy bajo.</li> </ul> Además, despliega los resultados y reporte completo.
+ printReporte(): void	Esta función muestra al final en la terminal un reporte con toda la información y resultados obtenidos.

## PARTE II. DISEÑO

### 2.1 Diagrama de clases de UML

Inserte aquí el diagrama exportado de la aplicación de modelación, como [www.Diagrams.net](http://www.Diagrams.net) vista en clase.



## PARTE III. PRUEBA

### 3.1 Plan de Prueba

Inserte aquí la tabla de casos de prueba para valores válidos, inválidos y valores limitados.

Prueba #1		
Leer fichero "info.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	50
	Edad de los Evaluados	26
	Total General de Graduados	29
	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	16
	Hombres Evaluados	25
	Total de Hombres Graduados	13
Salida		
	Zona	Kino
	Estado	Sonora
	Edad	26
	Total de Evaluados	50
	Porcentaje General de Graduados	58%
	Porcentaje Mujeres Graduadas	64%
	Porcentaje Hombres Graduados	52%
	Nivel de Alerta	Bajo

Prueba #2		
Leer fichero "info2.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	50
	Edad de los Evaluados	26
	Total General de Graduados	25
	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	25
	Hombres Evaluados	25
	Total de Hombres Graduados	0
Salida		
	Zona	Kino

	Estado	Sonora
	Edad	26
	Total de Evaluados	50
	Porcentaje General de Graduados	50%
	Porcentaje Mujeres Graduadas	100%
	Porcentaje Hombres Graduados	0%
	Nivel de Alerta	Bajo

Prueba #3		
Leer fichero "info3.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	50
	Edad de los Evaluados	30
	Total General de Graduados	25
	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	0
	Hombres Evaluados	25
	Total de Hombres Graduados	25
Salida		
	Zona	Kino
	Estado	Sonora
	Edad	30
	Total de Evaluados	50
	Porcentaje General de Graduados	50%
	Porcentaje Mujeres Graduadas	0%
	Porcentaje Hombres Graduados	100%
	Nivel de Alerta	Bajo

Prueba #4		
Leer fichero "info4.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	50
	Edad de los Evaluados	26
	Total General de Graduados	13
	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	8
	Hombres Evaluados	25



	Total de Hombres Graduados	5
Salida		
	Zona	Kino
	Estado	Sonora
	Edad	26
	Total de Evaluados	50
	Porcentaje General de Graduados	26%
	Porcentaje Mujeres Graduadas	32%
	Porcentaje Hombres Graduados	20%
	Nivel de Alerta	Alto

Prueba #5		
Leer fichero "info5.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	50
	Edad de los Evaluados	35
	Total General de Graduados	20
	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	11
	Hombres Evaluados	25
	Total de Hombres Graduados	9
Salida		
	Zona	Kino
	Estado	Sonora
	Edad	35
	Total de Evaluados	50
	Porcentaje General de Graduados	40%
	Porcentaje Mujeres Graduadas	44%
	Porcentaje Hombres Graduados	36%
	Nivel de Alerta	Alto

Prueba #6		
Leer fichero "info6.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	50
	Edad de los Evaluados	26
	Total General de Graduados	18

	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	10
	Hombres Evaluados	25
	Total de Hombres Graduados	8
Salida		
	Zona	Kino
	Estado	Sonora
	Edad	26
	Total de Evaluados	50
	Porcentaje General de Graduados	36%
	Porcentaje Mujeres Graduadas	40%
	Porcentaje Hombres Graduados	32%
	Nivel de Alerta	Alto

Prueba #7		
Leer fichero "info7.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	50
	Edad de los Evaluados	26
	Total General de Graduados	13
	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	8
	Hombres Evaluados	25
	Total de Hombres Graduados	5
Salida		
	Zona	Kino
	Estado	Sonora
	Edad	26
	Total de Evaluados	50
	Porcentaje General de Graduados	26%
	Porcentaje Mujeres Graduadas	32%
	Porcentaje Hombres Graduados	20%
	Nivel de Alerta	Alto

Prueba #8		
Leer fichero "info8.txt"		
Entrada	Datos	Cantidades

	Total de Evaluados	50
	Edad de los Evaluados	26
	Total General de Graduados	8
	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	6
	Hombres Evaluados	25
	Total de Hombres Graduados	2
Salida		
	Zona	Kino
	Estado	Sonora
	Edad	26
	Total de Evaluados	50
	Porcentaje General de Graduados	16%
	Porcentaje Mujeres Graduadas	24%
	Porcentaje Hombres Graduados	8%
	Nivel de Alerta	Grave

Prueba #9		
Leer fichero "info9.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	50
	Edad de los Evaluados	26
	Total General de Graduados	6
	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	2
	Hombres Evaluados	25
	Total de Hombres Graduados	4
Salida		
	Zona	Kino
	Estado	Sonora
	Edad	26
	Total de Evaluados	50
	Porcentaje General de Graduados	12%
	Porcentaje Mujeres Graduadas	8%
	Porcentaje Hombres Graduados	16%
	Nivel de Alerta	Grave

Prueba #10		
Leer fichero "info10.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	50
	Edad de los Evaluados	69
	Total General de Graduados	0
	Mujeres Evaluadas	25
	Total de Mujeres Graduadas	0
	Hombres Evaluados	25
	Total de Hombres Graduados	0
Salida		
	Zona	Kino
	Estado	Sonora
	Edad	69
	Total de Evaluados	50
	Porcentaje General de Graduados	0%
	Porcentaje Mujeres Graduadas	0%
	Porcentaje Hombres Graduados	0%
	Nivel de Alerta	Grave

Prueba #11		
Leer fichero "nombre_incorrecto.txt"		
Entrada	Datos	Cantidades
	Total de Evaluados	-
	Edad de los Evaluados	-
	Total General de Graduados	-
	Mujeres Evaluadas	-
	Total de Mujeres Graduadas	-
	Hombres Evaluados	-
	Total de Hombres Graduados	-
Salida		
	Zona	-
	Estado	-
	Edad	-
	Total de Evaluados	-
	Porcentaje General de Graduados	-

	Porcentaje Mujeres Graduadas	-
	Porcentaje Hombres Graduados	-
	Nivel de Alerta	-
	Mensaje de Error	Error al abrir nombre_incorr ecto.txt

## PARTE IV. PROGRAMA

### Evaluado.h

*// Definicion de la clase Evaluado*

*#ifndef Evaluado\_h*

*#define Evaluado\_h*

*#include <string>*

**using namespace std;**

**class Evaluado{**

*private: // Atributos privados*

string nombre;

**char** sexo;

string curp;

**char** graduado;

*public: // Metodos publicos*

Evaluado(); *// Constructor por defecto*

Evaluado(string, **char**, string, **char**); *// Constructor parametrizado*

**char** getSexo();

**char** getGraduado();

*~Evaluado(); // Destructor*

**};**

*#endif // Evaluado\_h*

**Evaluado.cpp***// Implementacion de la clase Evaluado**#include "Evaluado.h"**#include <cctype>**#include <string>**#include <iostream>***using namespace std;***Evaluado::Evaluado(){ // Constructor por defecto*

nombre = "";

sexo = '\_';

curp = "";

graduado=0;

}

*Evaluado::Evaluado( string cNombre, **char** cSexo, string cCurp, **char** cGraduado){ //*  
*Constructor parametrizado*

nombre = cNombre;

sexo = cSexo;

curp = cCurp;

graduado = cGraduado;

}

**char** Evaluado::getSexo() {**return** sexo;

}

**char** Evaluado::getGraduado(){

```
return graduado;  
}
```

```
Evaluado::~~Evaluado(){} // Destructor
```



**Poblacion.h***// Definicion de la clase Poblacion*

#ifndef Poblacion\_h

#define Poblacion\_h

#include "Evaluado.h"

#include &lt;iostream&gt;

#include &lt;vector&gt;

#include &lt;string&gt;

**class Poblacion**{private: *// Atributos privados***int** edad;

vector &lt; Evaluado &gt; listaEvaluados;

Evaluado evaluado;

public: *// Metodos publicos*Poblacion(); *// Constructor por defecto*Poblacion(**int**); *// Constructor parametrizado***int** getEdad();**void** setEvaluado(string, **char**, string, **char**);**void** setEdad(**int**);*// Funcionalidades***void** leerFichero(string);

```
int cantEvaluados();  
int contarGraduados();  
float porcentajeSexoM();  
float porcentajeSexoH();  
float porcentajeTotalGraduados();
```

```
~Poblacion(); // Destructor  
};
```

```
#endif // Poblacion_h
```

**Poblacion.cpp***// Implementacion de la clase Poblacion**#include "Poblacion.h"**#include <iostream>**#include <string>**#include <fstream>***using namespace std;****Poblacion::Poblacion()** *// Constructor por defecto*

edad = 25;

}

**Poblacion::Poblacion(int cEdad)** *// Constructor parametrizado*

edad = cEdad;

}

*// Metodos get***int Poblacion::getEdad()**{    **return** edad;

}

*// Metodos set***void Poblacion::setEvaluado(string cNombre, char cSexo, string cCarp, char cGraduado){**

Evaluado eva(cNombre, cSexo, cCarp, cGraduado);

listaEvaluados.push\_back(eva);

}

```
void Poblacion::setEdad(int cEdad){  
    edad = cEdad;  
}
```

*// Funcionalidades*

```
void Poblacion::leerFichero(string cFichero){ // Lee fichero  
    ifstream info;  
    info.open(cFichero);  
    if(!info.is_open()){  
        cout<<"Error al abrir "<<cFichero<<endl;  
        exit(EXIT_FAILURE);  
    }  
    int N;  
    info>>N;  
    info.ignore();  
    info>>edad;  
    info.ignore();  
  
    for(int i=0; i<N; i++){  
        string fNombre;  
        char fSexo;  
        string fCurp;  
        char fGraduado;  
  
        getline(info, fNombre);  
        info>>fSexo>>fCurp>>fGraduado;
```

```
info.ignore();

setEvaluado(fNombre, fSexo, fCarp, fGraduado);
}

info.close();
}

int Poblacion::cantEvaluados() { // Regresa la cantidad de evaluados dentro de la lista
    return listaEvaluados.size();
}

int Poblacion::contarGraduados() { // Cuenta y regresa la cantidad de evaluados graduados
    int sumaGrad = 0;

    for(int i=0; i<listaEvaluados.size(); i++){
        if(listaEvaluados[i].getGraduado() == '1'){
            sumaGrad+=1;
        }
    }

    return sumaGrad;
}

float Poblacion::porcentajeSexoM() { // Calcula y regresa el porcentaje de mujeres evaluadas graduadas
    int sumaMuj = 0;
```

```
int sumaMujGrad = 0;

for(int i=0; i<listaEvaluados.size(); i++){
    if(listaEvaluados[i].getSexo() == 'M'){
        sumaMuj+=1;
    }

    if((listaEvaluados[i].getSexo() == 'M') && (listaEvaluados[i].getGraduado() == 'I')){
        sumaMujGrad+=1;
    }
}

return ((float)sumaMujGrad/sumaMuj)*100;
}

float Poblacion::porcentajeSexoH() { // Calcula y regresa el porcentaje de hombres evaluados graduados
    int sumaHom = 0;
    int sumaHomGrad = 0;

    for(int i=0; i<listaEvaluados.size(); i++){
        if(listaEvaluados[i].getSexo() == 'H'){
            sumaHom+=1;
        }

        if((listaEvaluados[i].getSexo() == 'H') && (listaEvaluados[i].getGraduado() == 'I')){
            sumaHomGrad+=1;
        }
    }
}
```

```
    }  
}  
  
    return ((float)sumaHomGrad/sumaHom)*100;  
}  
  
float Poblacion::porcentajeTotalGraduados() { // Calcula y regresa el porcentaje de los evaluados graduados  
    int sumaTotGrad= 0;  
  
    for(int i=0; i<listaEvaluados.size(); i++){  
        if(listaEvaluados[i].getGraduado() == 'I'){  
            sumaTotGrad+=1;  
        }  
    }  
  
    return ((float)sumaTotGrad/listaEvaluados.size())*100;  
}  
  
Poblacion::~~Poblacion() {} // Destructor
```

**Zona.h**

*// Definicion de la clase Zona*

*#ifndef Zona\_h*

*#define Zona\_h*

*#include "Poblacion.h"*

*#include <string>*

*#include <iostream>*

**using namespace std;**

**class Zona{**

*private: // Atributos privados*

string nombreZona;

string estado;

Poblacion poblacion;

*public: // Metodos publicos*

*Zona(); // Constructor por defecto*

*Zona(string, string); // Constructor parametrizado*

string getNombreZona();

string getEstado();

*// Funcionalidades*

**void** leerDatosPoblacion(string);

string nivelAlerta();



```
void printReporte();
```

```
~Zona(); // Destructor
```

```
};
```

**Zona.cpp***// Implementacion de la clase Zona**#include "Zona.h"**#include <iostream>**#include <string>**#include <fstream>***using namespace std;***Zona::Zona(){ // Constructor por defecto*

nombreZona = "NA";

estado = "NA";

}

*Zona::Zona(string cNombreZona, string cEstado){ // Constructor parametrizado*

nombreZona = cNombreZona;

estado = cEstado;

}

*// Metodos get*

string Zona::getNombreZona(){

**return** nombreZona;

}

string Zona::getEstado(){

**return** estado;

}

*// Funcionalidades*

**void** Zona::leerDatosPoblacion(string fichero){ *// Lee la informacion del fichero de la clase Poblacion*

    poblacion.leerFichero(fichero);

}

string Zona::nivelAlerta(){ *// Calcula y regresa el nivel de alerta*

    string grave= "Grave";

    string alto= "Alto";

    string bajo= "Bajo";

    string muyBajo= "Muy Bajo";

**if**(poblacion.porcentajeTotalGraduados() < 20){

**return** grave;

    }

**else if**((poblacion.porcentajeTotalGraduados() >= 20) &&  
(poblacion.porcentajeTotalGraduados() < 50)){

**return** alto;

    }

**else if**((poblacion.porcentajeTotalGraduados() >= 50) &&  
(poblacion.porcentajeTotalGraduados() < 80)){

**return** bajo;

    }

**else**{

**return** muyBajo;

    }

}

```
void Zona::printReporte(){ // Imprime en pantalla el reporte con los resultados

    cout<<"Resultados de la evaluacion: \n"<<endl;

    cout<<"Zona: "<<getNombreZona()<<endl;

    cout<<"Estado: "<<getEstado()<<endl;

    cout<<"Grupo de edad: "<<poblacion.getEdad()<<endl;

    cout<<"Numero de personas evaluadas: "<<poblacion.cantEvaluados()<<endl;

    cout<<"\nEl "<<poblacion.porcentajeTotalGraduados()<<" % "<<"del total de los evaluados  
se ha graduado de sus estudios universitarios.\n"<<endl;

    cout<<"  Del total de las mujeres evaluadas, el "<<poblacion.porcentajeSexoM()<<" %  
"<<"se ha graduado de sus estudios universitarios."<<endl;

    cout<<"  Del total de los hombres evaluados, el "<<poblacion.porcentajeSexoH()<<" %  
"<<"se ha graduado de sus estudios universitarios."<<endl;

    cout<<"\nEl nivel de alerta de la zona es: "<<nivelAlerta()<<endl;

}

Zona::~Zona(){} // Destructor
```

**main.cpp***// Programa principal**#include "Zona.h"**#include <iostream>**#include <string>**#include <fstream>***using namespace std;****int main(){**

Zona zona1("Kino", "Sonora");

zona1.leerDatosPoblacion("info.txt");

zona1.printReporte();

**return 0;****}**

**Referencias:**

- AP. (2021). 5.2 millones de alumnos dejaron la escuela por pandemia de COVID en México.  
Recuperado de  
<https://www.elfinanciero.com.mx/nacional/2021/08/13/52-millones-de-alumnos-dejaron-la-escuela-por-pandemia-de-covid-en-mexico/>
- García, A. (2018). Educación en México: insuficiente, desigual y la calidad es difícil de medir.  
Recuperado de  
<https://www.eleconomista.com.mx/politica/Educacion-en-Mexico-insuficiente-desigual-y-la-calidad-es-dificil-de-medir-20181225-0028.html>
- INEE. (22 de abril de 2018). La desigualdad es el principal problema educativo de México: Sylvia Schmelkes. Instituto nacional para la Evaluación de la Educación.  
<https://www.inee.edu.mx/la-desigualdad-es-el-principal-problema-educativo-de-mexico-sylvia-schmelkes/>