



INSTITUTO POLITECNICO NACIONAL

**ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
UNIDAD ZACATENCO
INGENIERIA EN COMUNICACIONES Y ELECTRONICA**

Ingeniería de software

Algoritmos Genéticos

**Toledo Hernandez Ulises Gabriel
2020302316**

8CM12

Prof. Mancilla León Armando

Planeación

El propósito del siguiente proyecto es poder exponer los puntos clave mediante un visualizador de archivos PDF en el que se presentará toda la información de cada uno de los subtemas correspondientes, así mismo en el menú se podrá desplegar la firma y a su vez salir del descriptor.

Análisis

El proyecto se enfoca principalmente en mostrar de forma visual toda la investigación sobre los *Algoritmos Genéticos* con ayuda de un descriptor de Windows Forms usando algunas herramientas de prácticas pasadas. Se enfoca en mostrar de forma más atractiva la información del tema sin llegar a ser abrumante para el usuario a través de un *menuStrip*.

Algoritmos genéticos

Durante los últimos treinta años ha crecido el interés en sistemas de resolución de problemas basados en los principios de evolución y herencia.

Los Algoritmos Genéticos (AGs) son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización. Están basados en el proceso genético de los organismos vivos. A lo largo de las generaciones, las poblaciones evolucionan en la naturaleza de acorde con los principios de la selección natural y la supervivencia de los más fuertes, postulados por Darwin (1859). Por imitación de este proceso, los Algoritmos Genéticos son capaces de ir creando soluciones para problemas del mundo real. La evolución de dichas soluciones hacia valores óptimos del problema depende en buena medida de una adecuada codificación de estas.

Un algoritmo genético tipo cuenta con un bucle principal, en el que se realizan las acciones características más importantes. La siguiente enumeración representa ordenadamente cómo actúan:

1. Se genera una población inicial de soluciones aleatoriamente obtenidas.
2. Se evalúan las soluciones con la función de aptitud o evaluación.
3. Se comprueba si se ha obtenido una solución óptima (o suficientemente buena) o si se cumple alguna otra condición de parada. En caso afirmativo, termina. En caso negativo, continúa en el paso cuatro.
4. Se seleccionan n individuos (dependiendo del método de selección)
5. Se cruzan los individuos seleccionados y se obtienen nuevos individuos. a. Los nuevos individuos sufren o no mutación de manera aleatoria.

6. Se reemplaza la población con los nuevos individuos obtenidos y se vuelve al paso número dos. De esta manera, aunque depende del método de selección, lo deseable es que se elija con mayor probabilidad a los individuos más aptos, de manera que al combinar las soluciones en la fase de cruce, se obtengan soluciones que probablemente sean igual de aptas, o incluso más aptas.

Para un problema particular deben tener los siguientes 5 componentes:

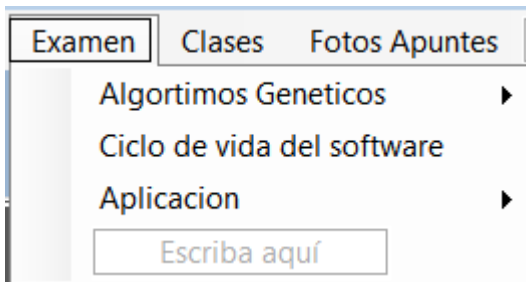
- Una representación genética de las soluciones potenciales.
- Una forma de crear la población inicial.
- Una función de evaluación que juegue el rol del ambiente clasificando los individuos en función de su adaptación.
- Operadores genéticos que alteren la composición de los hijos.
- Valores de los parámetros para las distintas variables que usa el algoritmo (tamaño de la población, probabilidad de aplicación de los operadores genéticos etc.)

Para definir un algoritmo genético debemos entonces definir los siguientes componentes:

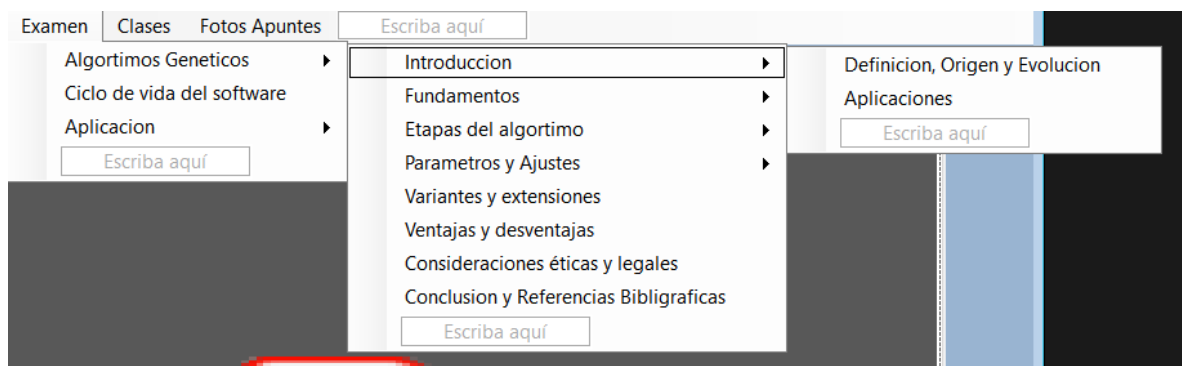
- Población de individuos: cada uno de ellos representa una posible solución.
- Función de aptitud de los individuos: indica que tan bueno es el individuo en la población.
- Función objetivo: criterio de parada del algoritmo.
- Selección y tasa de selección: de qué forma seleccionaremos los mejores individuos.
- Cruzamiento y tasa de cruzamiento: cómo combinaremos la información de dos individuos.
- Mutación y tasa de mutación: modificar parte de un individuo para introducir variedad en la población.

Diseño

Para el diseño del descriptor se tomó en cuenta la herramienta *MenuStrip* que nos facilita la organización de nuestras opciones a manipular, en este caso para el Examen desplegamos el tema, el ciclo de vida del tema y la aplicación (transformada rápida de Fourier).

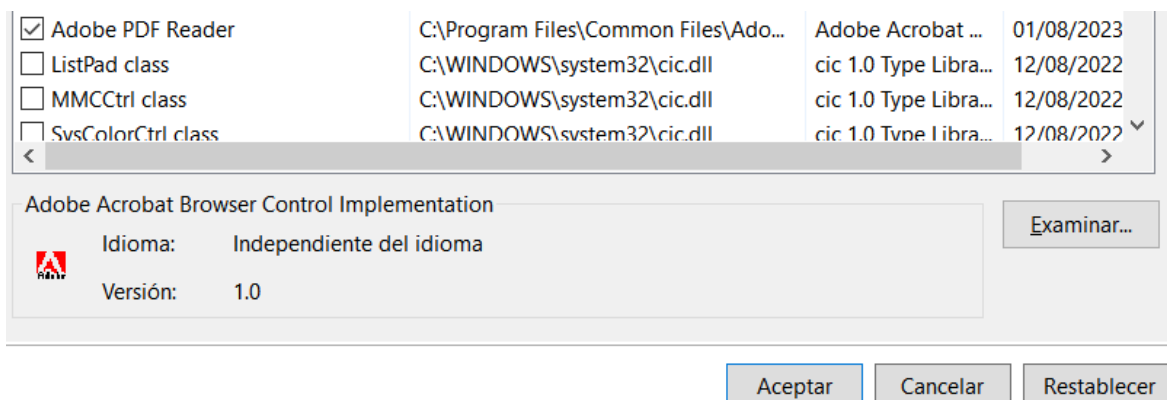
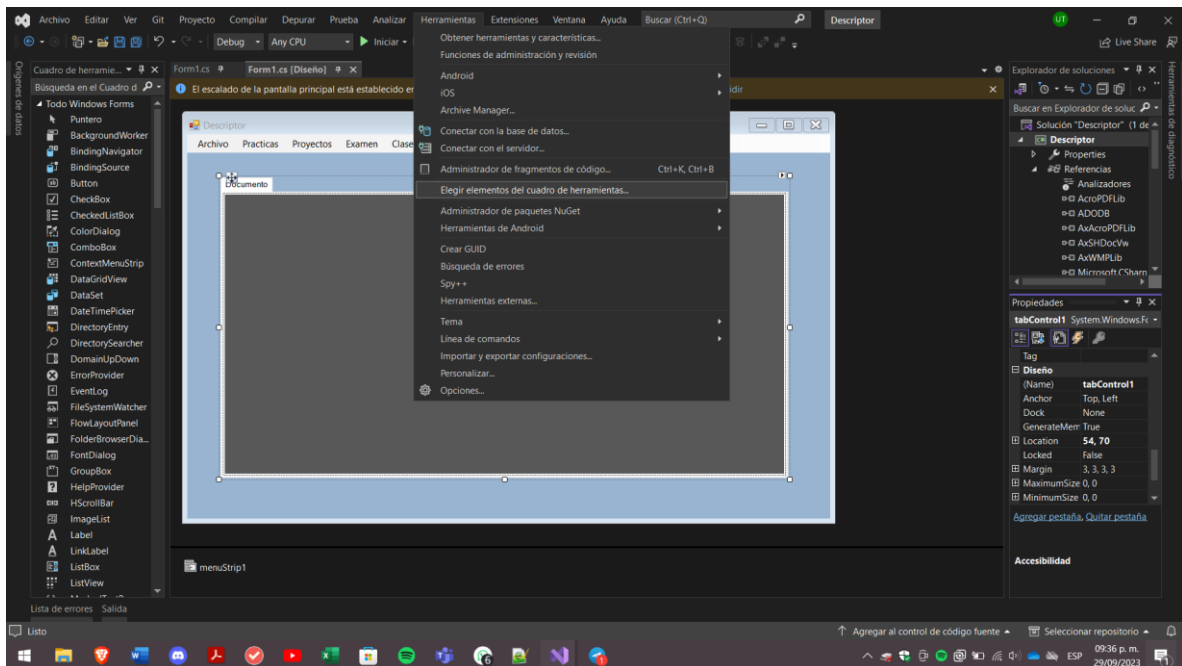


A su vez se dividió en mas submenús cada tema para que sea mas fácil el acceso a cada uno de los puntos que se quiere enfocar el tema a tratar y no sea tan engorroso la búsqueda de estos.



En el descriptor con ayuda de la herramienta *TabControl* Declaramos el apartado de documento donde se desplegarán los archivos correspondientes a cada tema y subtema en cuestión.

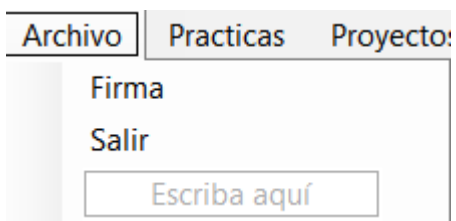
Para esto primero en *Herramientas-Elegir elementos del cuadro de herramientas-Componentes COM-ADOBE READER-Aceptar*. Y así en el cuadro de herramientas aparecerá la herramienta *Adobe Reader*.



Es así que podremos agregar Adobe Reader dentro del TabControl. Se modifica el tamaño de tal forma que abarque todo el TabControl y sea mas fácil la visualización de los documentos.



En archivo se tienen las opciones para desplegar tanto la firma como para salir del descriptor.



Implementación

Para el despliegue de los archivos en el TabControl se realiza el siguiente código en el con ayuda de un objeto mandamos a llamar a cada uno de los .pdf de los temas.

```
//Investigacion
1 referencia
private void definicionToolStripMenuItem_Click(object sender, EventArgs e)
{
    OpenFileDialog arch1 = new OpenFileDialog();
    arch1.FileName = "Inv\\AG1.pdf";
    PDF1.LoadFile(arch1.FileName);
}
```

De igual forma mandamos a llamar al archivo correspondiente en el ciclo de vida.

```
private void cicloToolStripMenuItem_Click(object sender, EventArgs e)
{
    OpenFileDialog arch1 = new OpenFileDialog();
    arch1.FileName = "Ciclodevida.pdf";
    PDF1.LoadFile(arch1.FileName);
}
```

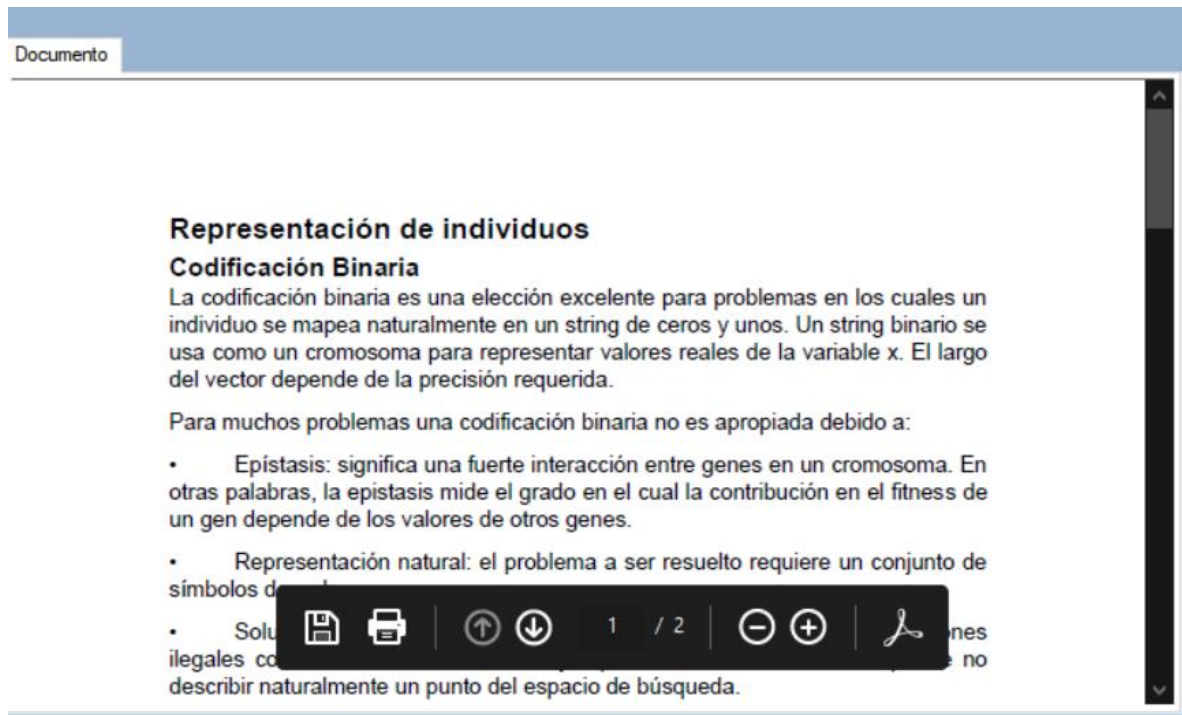
Se despliega mediante la siguiente función la firma, además con "Close();" cerramos el descriptor.

```
private void firmaToolStripMenuItem_Click(object sender, EventArgs e)
{
    System.Diagnostics.Process.Start("FIRMA.exe");
}
```

```
1 referencia
private void salirToolStripMenuItem_Click(object sender, EventArgs e)
{
    Close();
}
```

Procesamiento

El descriptor mostrara el archivo según la opción de tema que se elija dentro del TabControl en la sección de documento e interactuar con el. Además si seleccionamos otro tema, se cerrará el que se esta mostrando y se abrirá el nuevo pdf.



A su vez en archivo, se puede desplegar la firma y cerrarla con el botón en color rojo.



Mantenimiento

Añadir nuevas funcionalidades para hacerlo mas ameno a las necesidades del usuario, a su vez este podrá verse más profesional y estética, así como atractivo para la persona que lo use. Mejorar el rendimiento del descriptor a partir de las nuevas funcionalidades puede presentar ventajas con respecto a otros proyectos.