

Proyecto 1 LFP

Descripción breve

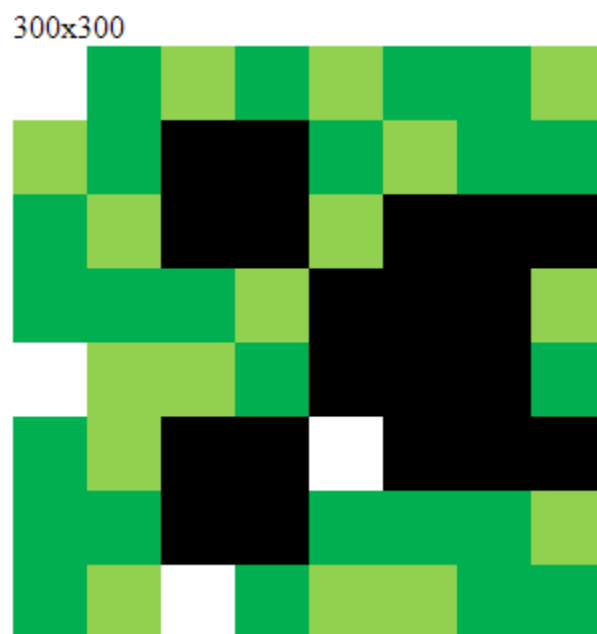
Encontrara un manual el cual le ayudara a comprender de mejor manera el código en caso se necesite ser editado o modificado en un futuro

Alexander Mejia
201900576

Descripción

La empresa *Bitxelart* se dedica a la elaboración de imágenes digitales en estilo **pixel art**, para realizar estas piezas digitales un empleado toma una imagen que utilizará como referencia y la divide en una cuadrícula, anotando en un **archivo de texto**, con extensión **pxla**, las características de la misma (color, dimensiones, etc), luego este archivo es trasladado a este programa de python para optimizar digital que utilizando el lenguaje de marcado **HTML**, dibuja la imagen solicitada..

Un ejemplo de estas imágenes es el siguiente:



El archivo se compone de 7 secciones, la estructura es la siguiente:

TITULO="Pokebola";

ANCHO=300;

ALTO=300;

FILAS=12;

COLUMNAS=12;

CELDAS = {

[0,0,FALSE,#000000],

[0,1,FALSE,#000000],

[3,3,FALSE,#000000],

[3,4,TRUE,#000000],

[3,5,TRUE,#000000],

[3,6,TRUE,#000000],

[3,7,TRUE,#000000],

[4,1,FALSE,#000000]

};

FILTROS = MIRRORX;

@@@@

TITULO="Estrella";

ANCHO=300;

ALTO=300;

FILAS=4;

```
COLUMNAS=4;

CELDAS = {
[0,0,FALSE,#000000],
[1,1,FALSE,#000000],
[3,3,FALSE,#000000],
[2,1,FALSE,#000000]
};

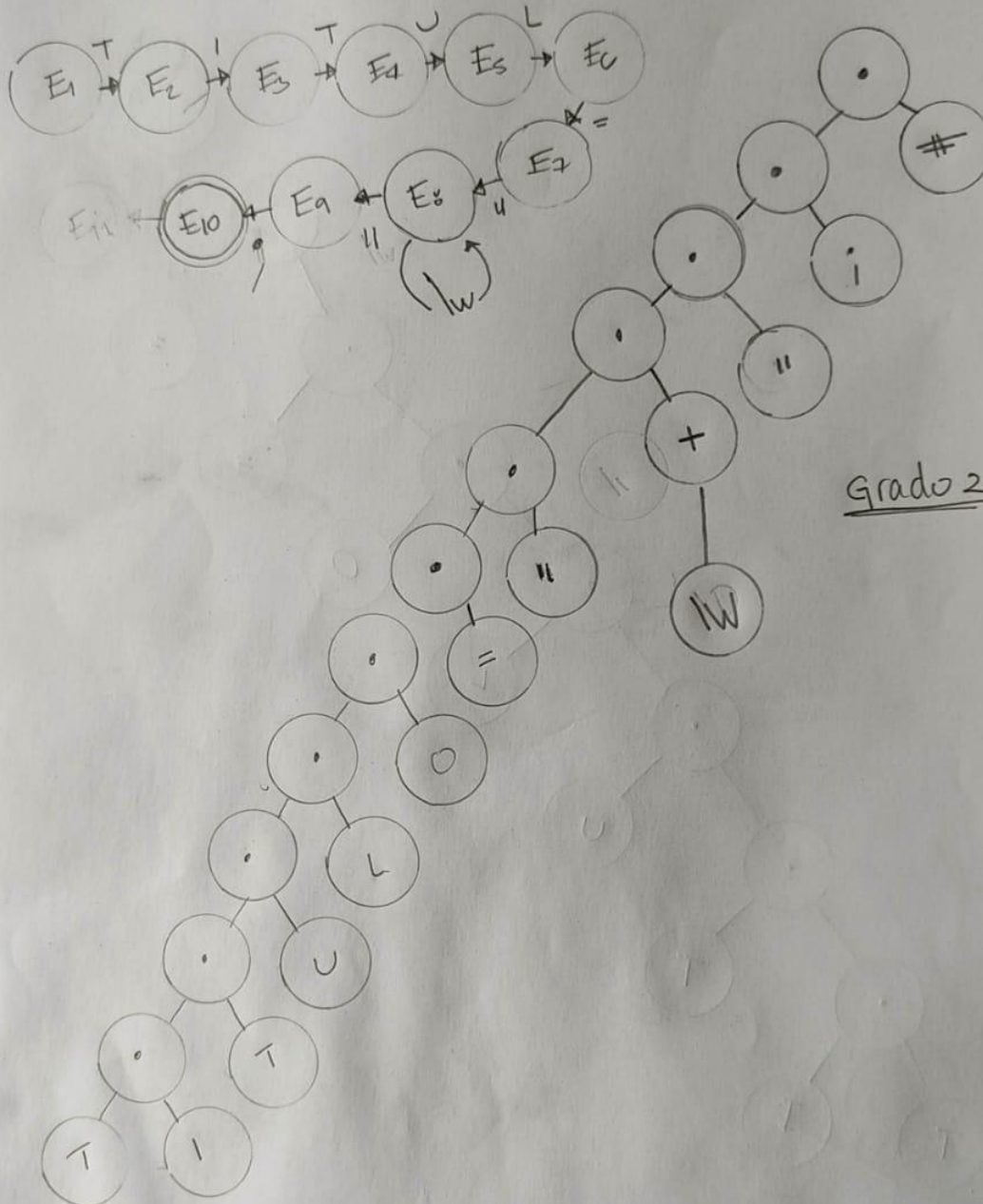
FILTROS = MIRRORX,MIRRORY,DOUBLEMIRROR;
```

1. TITULO:

Esta sección indica el título de la imagen que se generará, se conforma de la palabra reservada TITULO seguida de un signo igual (=) y una cadena de texto encerrada en comillas dobles (") y finaliza con punto y coma (;).

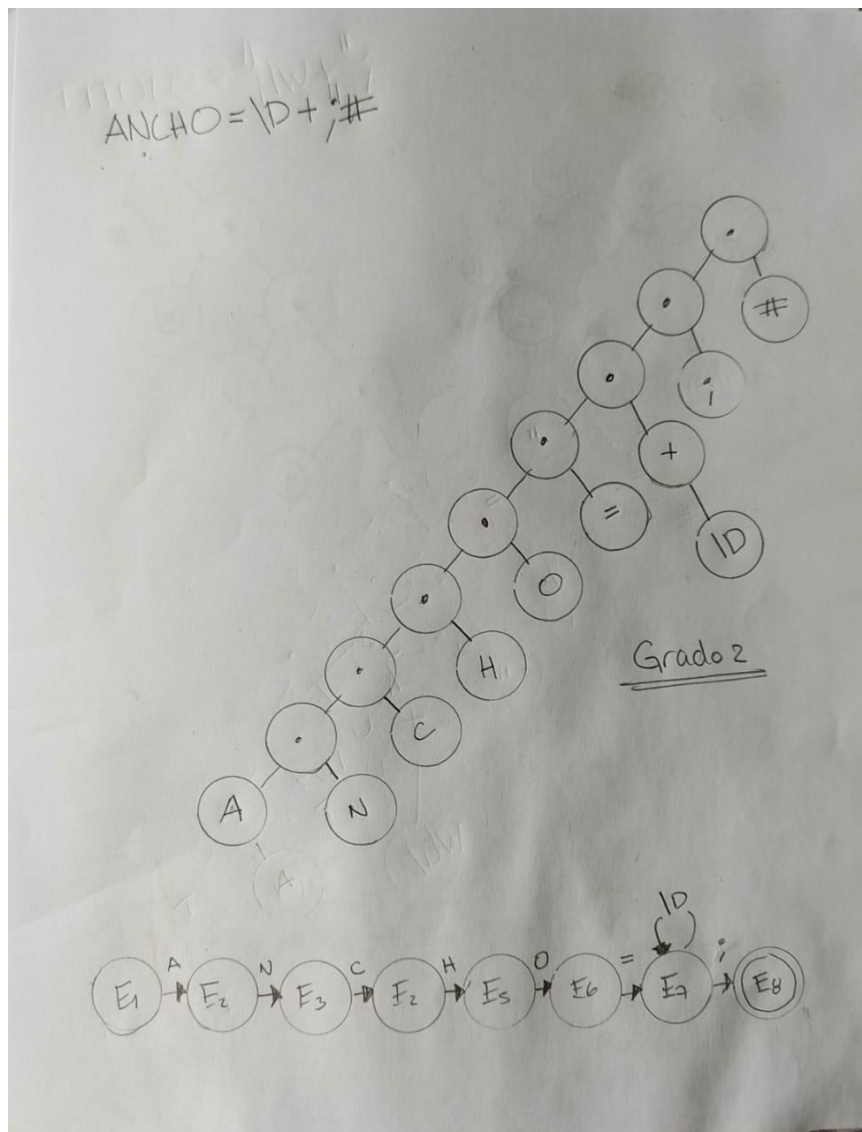
A continuación se muestra el listado de tokens (expresión regular) y su respectivo diagrama de árbol y el diagrama del autómata.

TITULO = "//W+" ;



2. ANCHO:

Esta sección indica el ancho en píxeles que tendrá la imagen, la sección se conforma de la palabra reservada ANCHO seguida de un signo igual (=), como valor tomará un número entero positivo y finaliza con un punto y coma (;). A continuación se muestra el listado de tokens (expresión regular) y su respectivo diagrama de árbol y el diagrama del autómata.



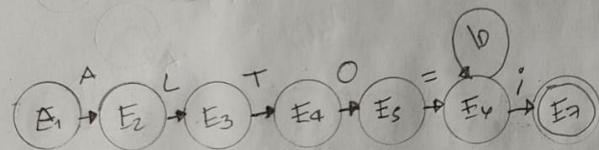
3. ALTO:

ALTO = 1D+;#

Grado 2

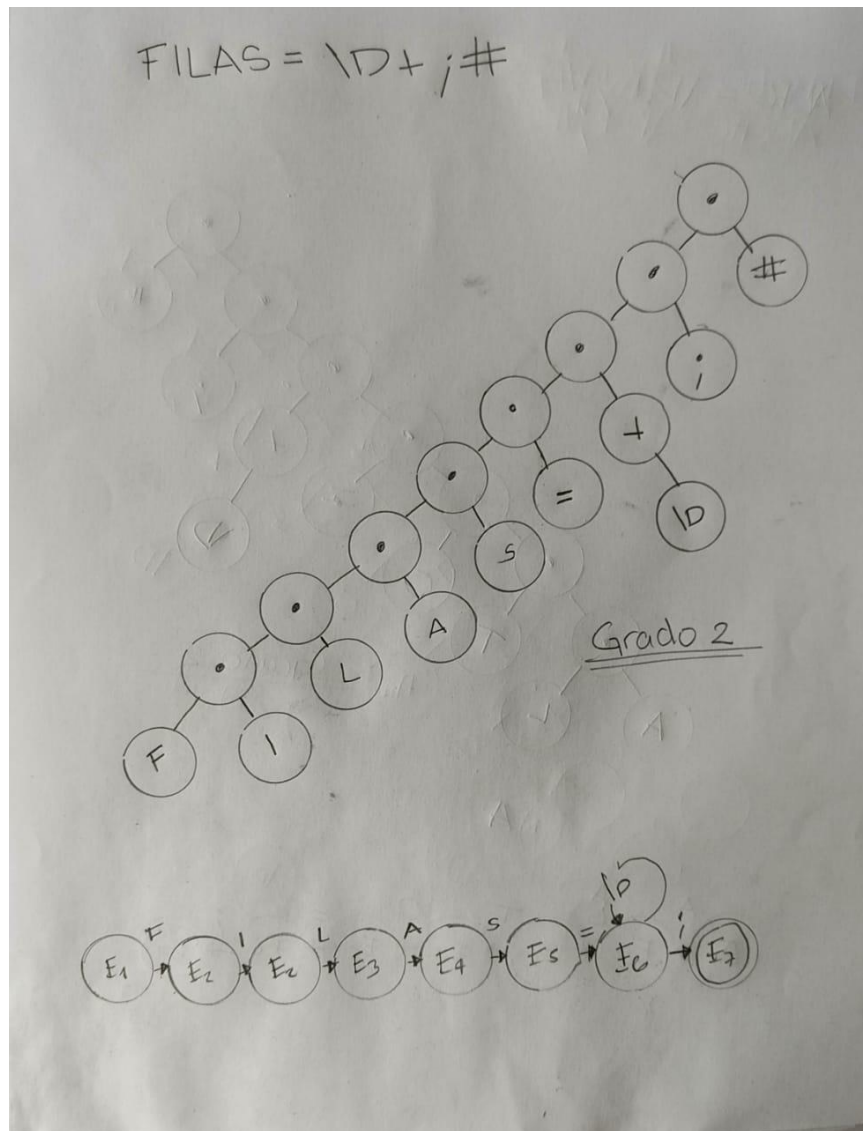
Diagrama de estados:

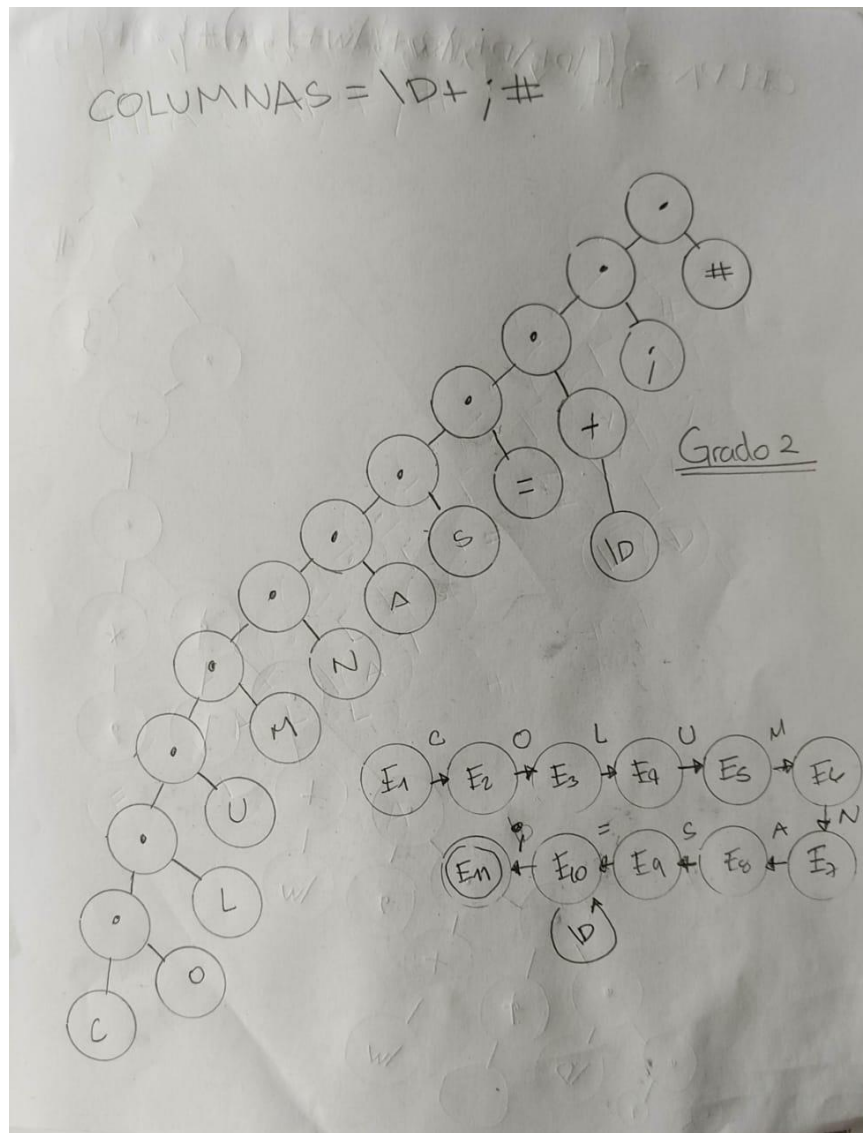
```
graph LR
    E1((E1)) -- A --> E2((E2))
    E2 -- L --> E3((E3))
    E3 -- T --> E4((E4))
    E4 -- O --> E5((E5))
    E5 -- = --> E6((E6))
    E6 -- i --> E7(((E7)))
    E6 -- D --> E6
```



4. FILAS:

Esta sección indica el número de filas en las que se divide la imagen, la sección se conforma de la palabra reservada FILAS seguida de un signo igual (=), como valor tomará un número entero positivo y finaliza con un punto y coma (;). A continuación se muestra el listado de tokens (expresión regular) y su respectivo diagrama de árbol y el diagrama del autómata.





6. CELDAS:

En esta sección se definen los colores que tomará cada uno de los “píxeles” en los que está dividido la imagen, la sección inicia con la palabra reservada CELDAS seguida de un signo igual (=), como valor tomará una lista de elementos que estará encerrada en un par de llaves y cada elemento separado por una coma (,).

Los elementos de esta lista se encuentran encerrados en un par de corchetes y dentro de estos están dos valores separados por coma (,). El primer elemento es un valor entero positivo que representa la posición de la celda en el eje X, el segundo elemento es un valor entero positivo que representa la posición de la celda en el eje Y, el tercer valor es una cadena que representa un booleano (TRUE o FALSE) y por último un color representado en forma hexadecimal. Ejemplos:

- [0,0,TRUE,#FA0000]

True nos indica que la celda sí será pintada del color indicado, el color es #FA0000 que equivale al color rojo.

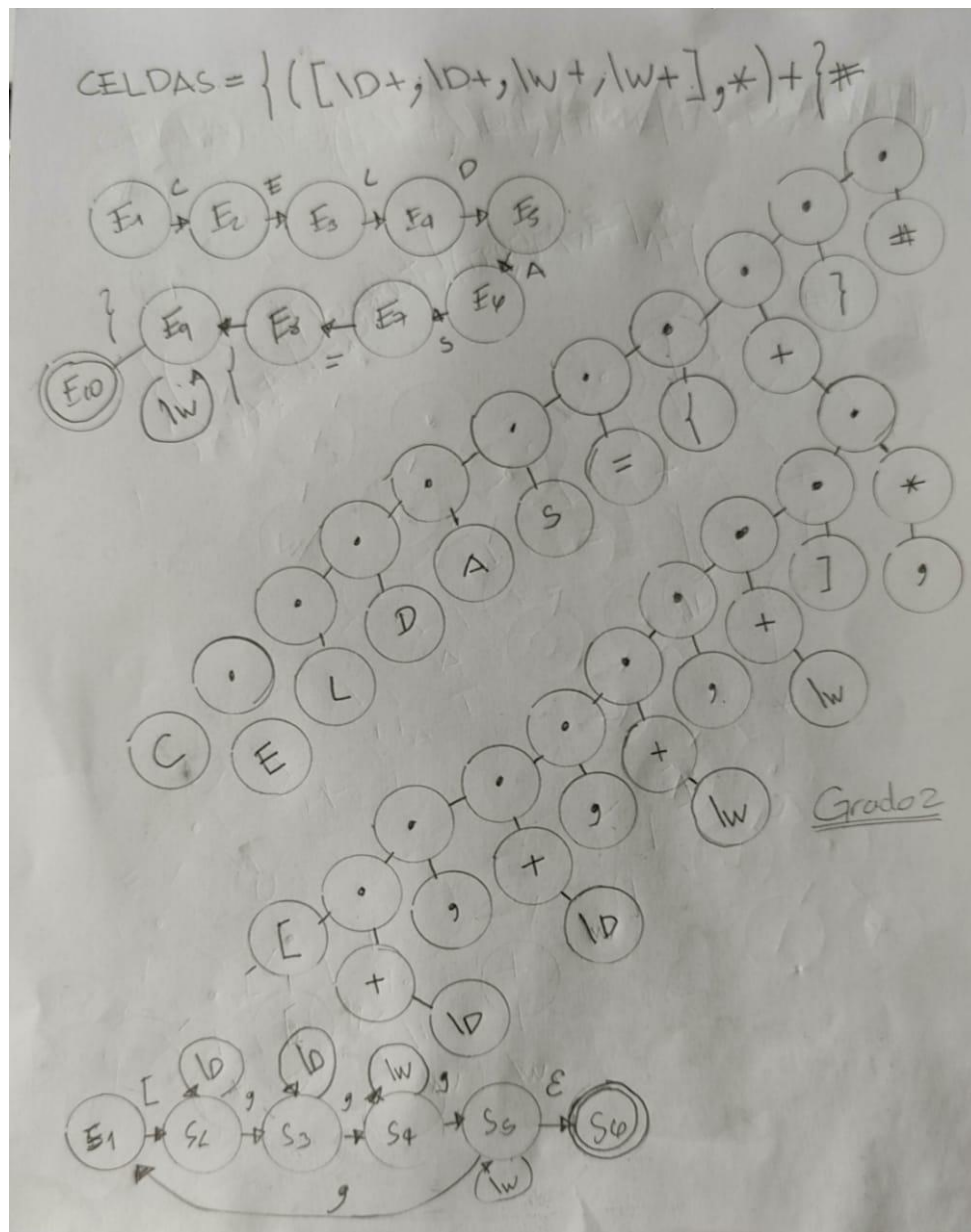
- [0,1,FALSE,#FA0000]

False nos indica que la celda no será pintada del color indicado, es decir el color es ignorado y la celda no se pintará.

** Si una posición no aparece no se pintará de ningún color, lo que es equivalente a que la celda tuviera el parámetro FALSE en ella.

La sección finaliza con un punto y coma(;).

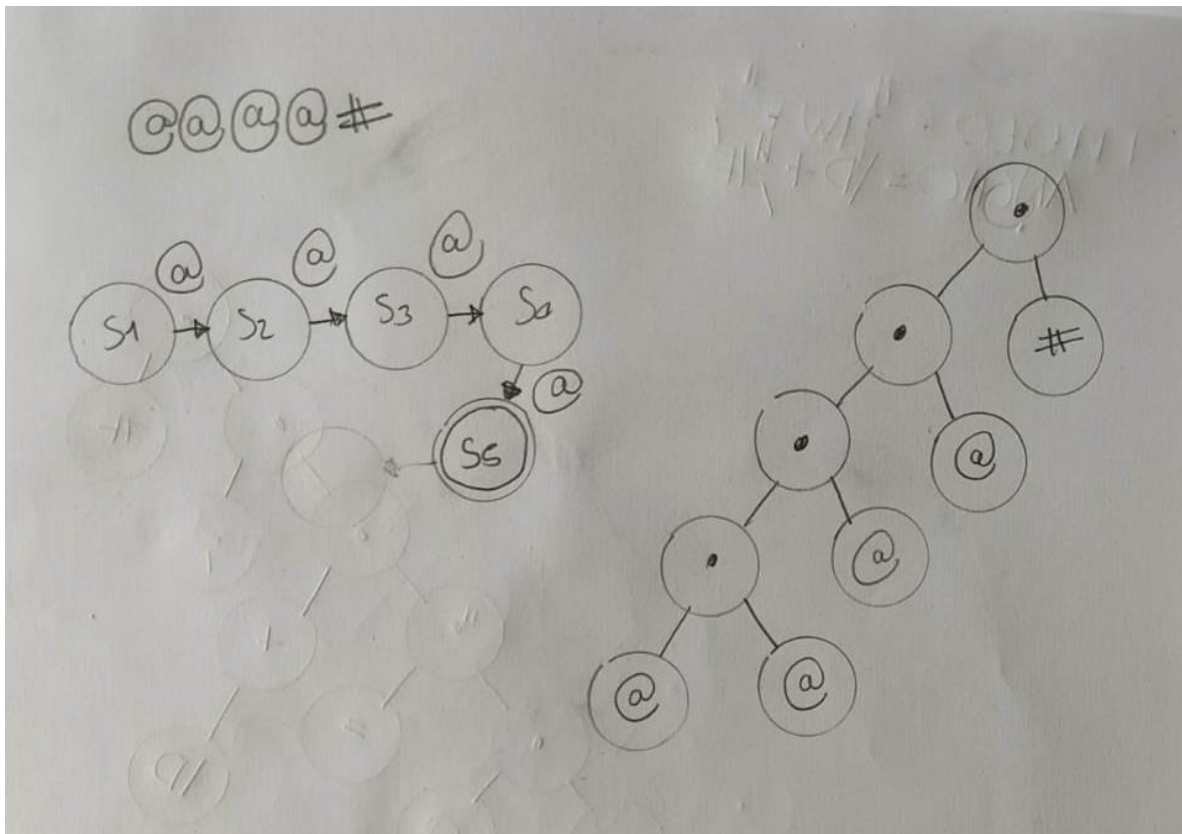
A continuación se muestra el listado de tokens (expresión regular) y su respectivo diagrama de árbol y el diagrama del autómata.

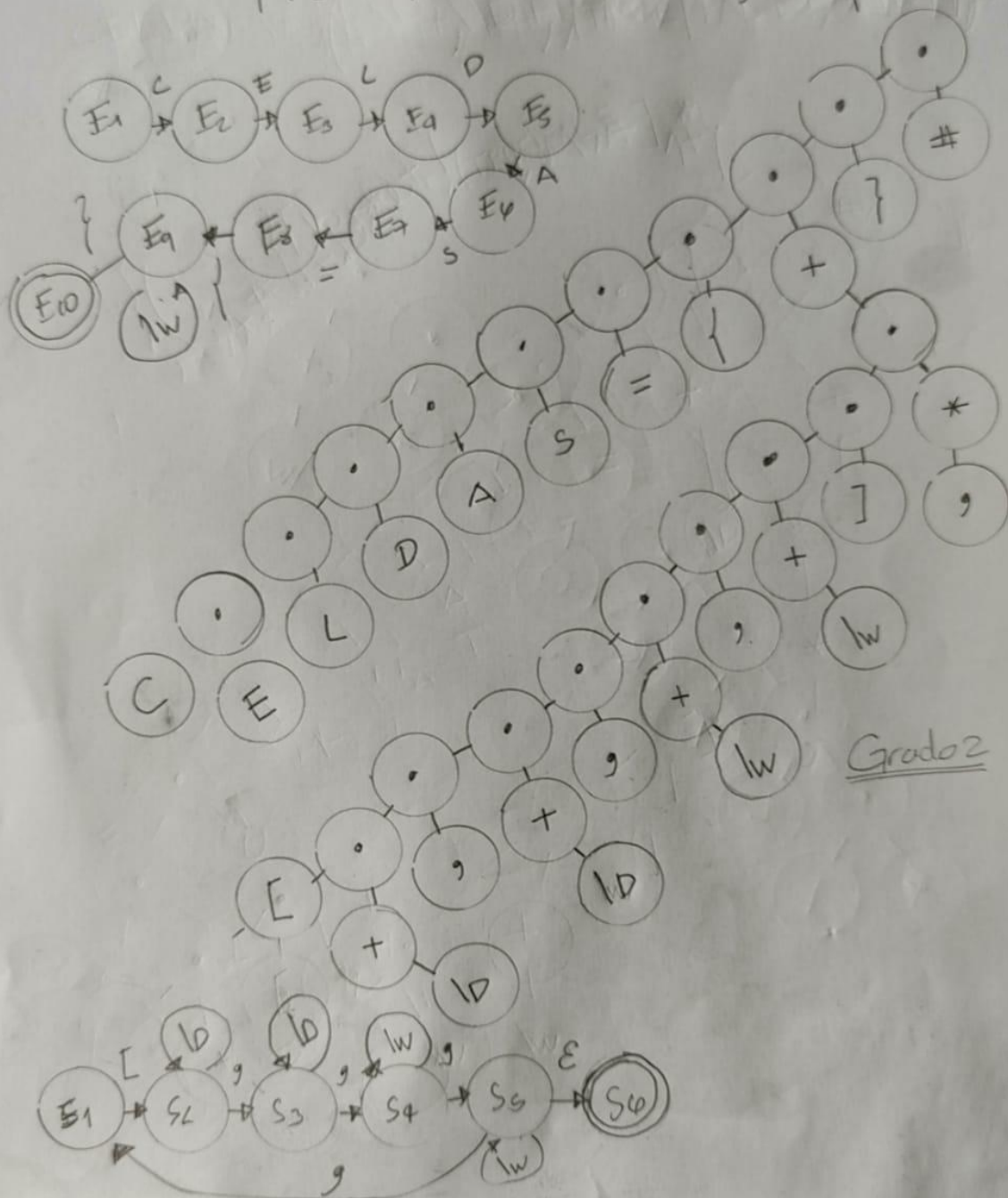


Separador

Al finalizar la definición de una imagen puede aparecer un separador que consiste en una cadena de cuatro arrobas (@@@@), lo cual nos indica que en cada archivo es posible definir 'N' imágenes.

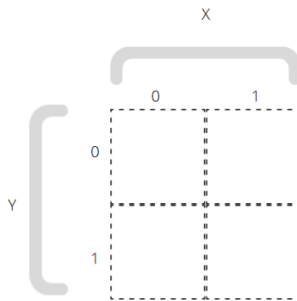
A continuación se muestra el listado de tokens (expresión regular) y su respectivo diagrama de árbol y el diagrama del autómata.



$$CELDA5 = \{ ([1D+, 1D+, 1W+, 1W+], *) + \} \#$$


Tamaño de la imagen y de los pixeles

Para calcular el tamaño de cada pixel se tomo en cuenta el tamaño en pixeles y se dividio entre las filas y columnas es decir si el ancho es de 300 y se tiene 6 columnase divide $300/6$ y obtenemos el tamaño del pixel haciendo que quede del tamaño que se nos pide



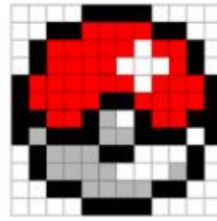
7. FILTROS:

Esta función indica los filtros que podrán ser aplicados a esta imagen, la sección inicia con la palabra reservada *FILTROS* seguida de un signo igual (=), seguida de una lista de palabras reservadas separadas por una coma (,) la sección finaliza con un punto y coma (;). Las palabras reservadas pueden ser:

- MIRRORX: Este filtro gira horizontalmente la imagen original.
- MIRRORY: Este filtro gira verticalmente la imagen original.
- DOUBLEMIRROR: Este filtro gira horizontal y verticalmente la imagen original



Imagen original



MIRRORX



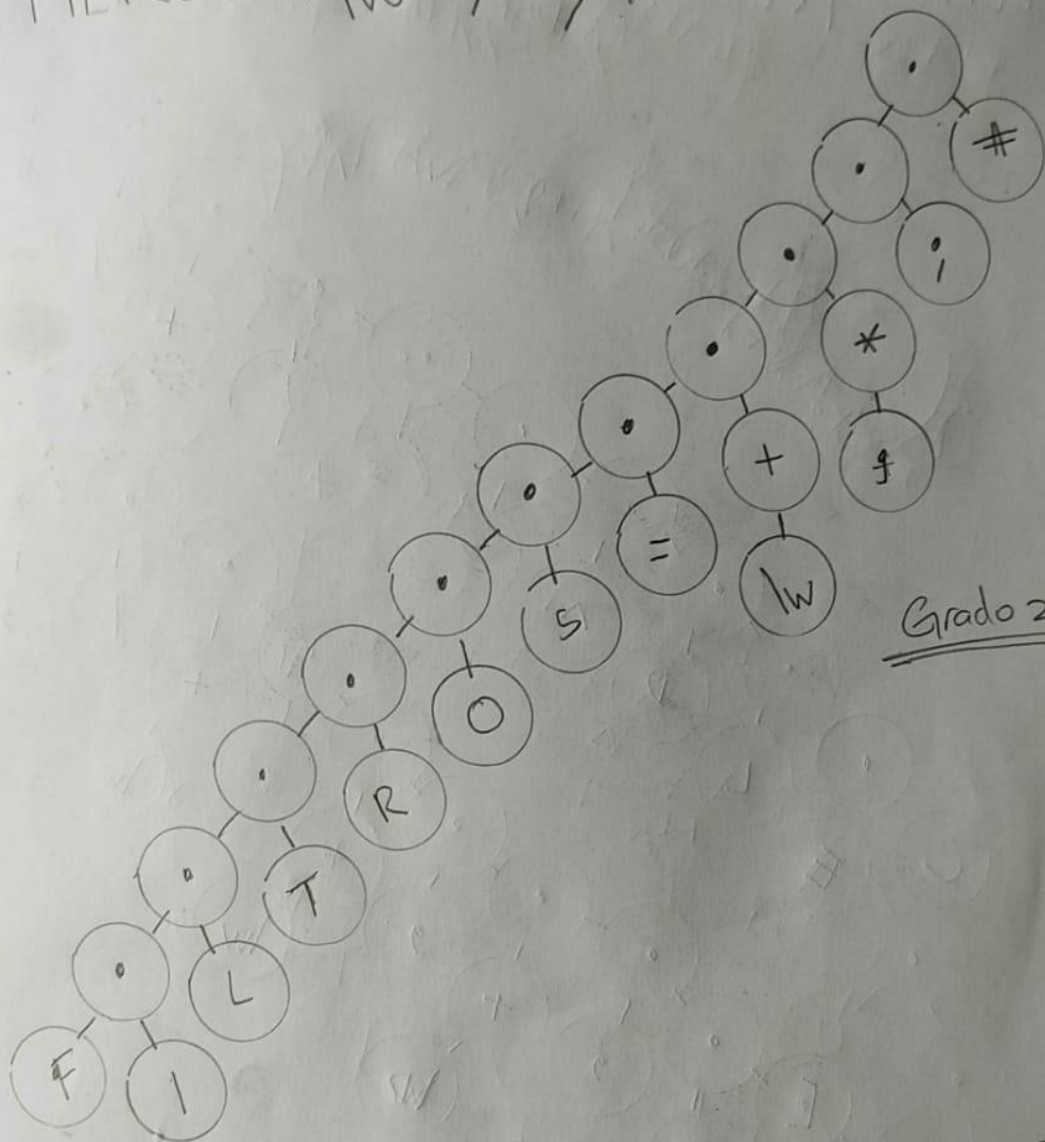
MIRRORY



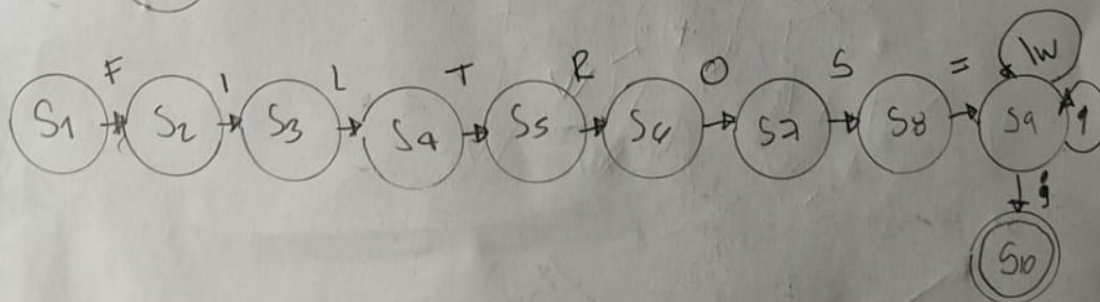
DOUBLEMIRROR

A continuación se muestra el listado de tokens (expresión regular) y su respectivo diagrama de árbol y el diagrama del autómata.

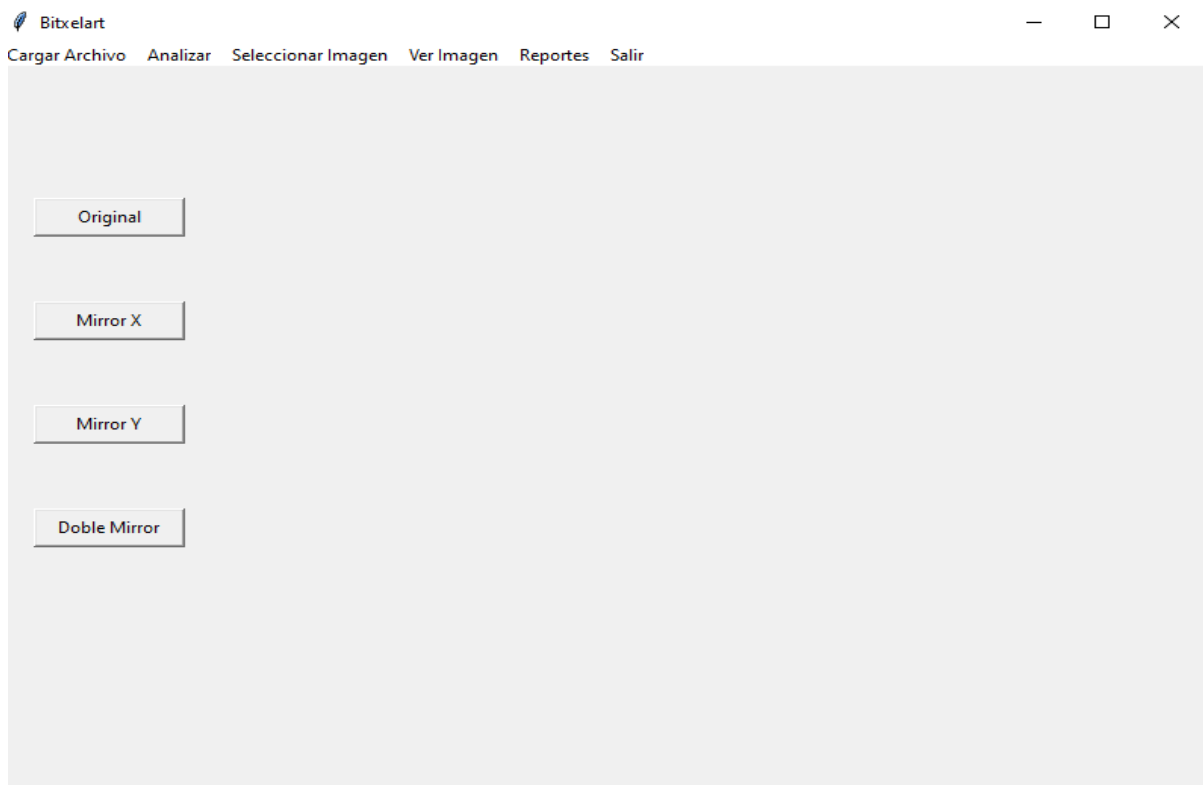
FILTROS = (\w +), * ; #



Grado 2



Intefaz Grafica

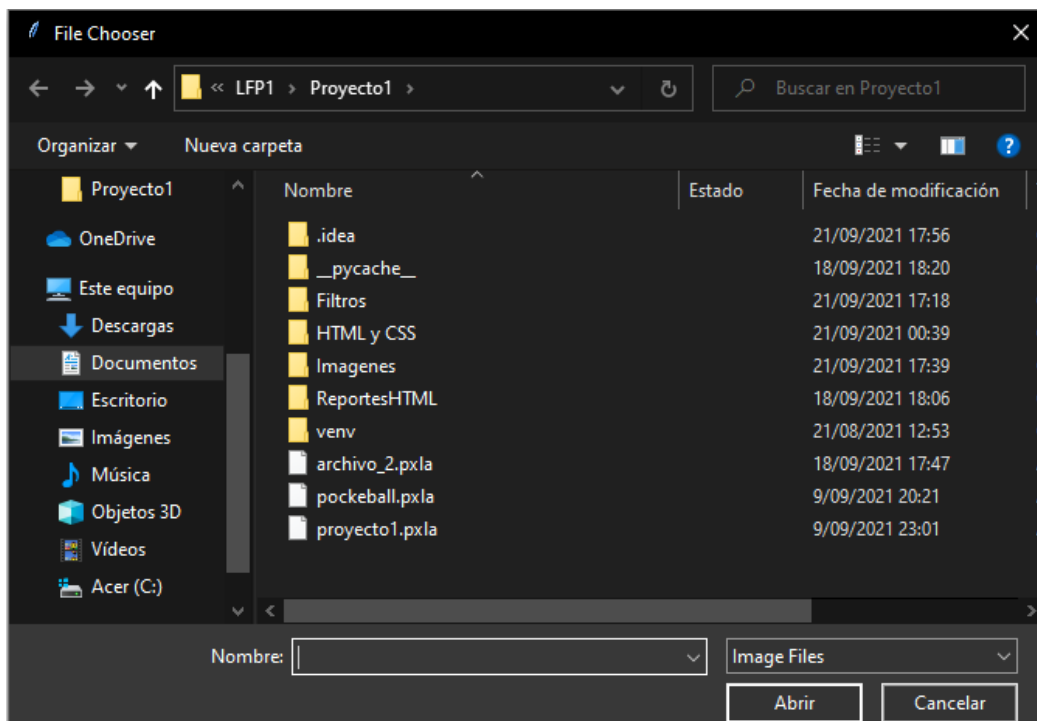


Funciones del sistema

La aplicación cuenta con un interfaz gráfica, en la cual se poseen las siguientes opciones:

1. Cargar archivo:

Muestra una ventana emergente que permite al usuario seleccionar un archivo pxla y cargarlo a memoria.



Este programa solo reconoce archivos .pxla

2. Analizar archivo:

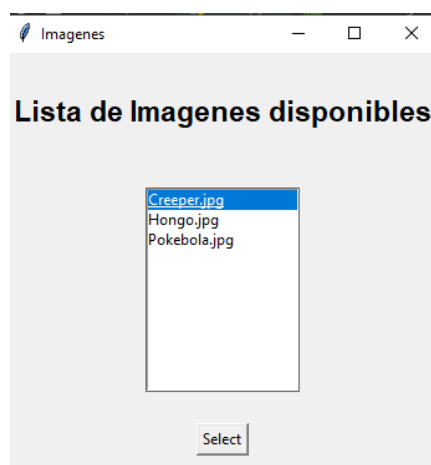
Analiza el archivo cargado mediante un *autómata definido e implementado por el estudiante* y genera los archivos HTML tanto de la imagen original como de las imágenes con filtro. los datos obtenidos durante la ejecución del reporte.



4. Seleccionar imagen:

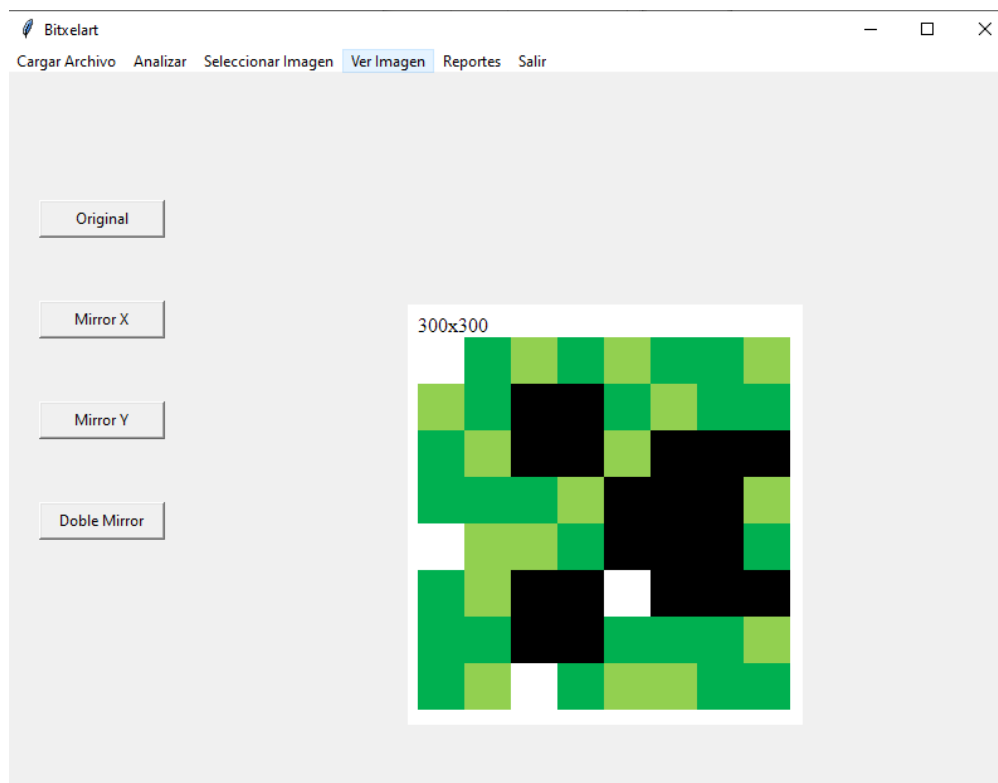
El programa cuenta con una lista de las imágenes analizadas en el archivo, se deberá poder seleccionar una imagen para poder visualizarla junto con los filtros aplicados en ella.

Se desplegara una nueva ventana con las fotos que leyó del archivo



5. Ver imagen:

El programa cuenta con un apartado para poder visualizar las imágenes generadas. Para poder visualizar una imagen debe de seleccionarla antes en el botón seleccionar imágenes.



a. Original: Muestra la imagen original sin aplicar ningún filtro.

b. MirrorX: Muestra la imagen girada horizontalmente.

c. MirrorY: Muestra la imagen girada verticalmente

d. DoubleMirror: Muestra la imagen girada horizontal y verticalmente.

Si se selecciona un filtro que no ha sido generado debe notificar al usuario.

6. Salir:

Termina la ejecución de la aplicación.

```
Process finished with exit code 0
```