

PROYECTO 2

Integrantes:

- Diego Andres Alonzo Medinilla 20172
- Samuel Argueta 211024

Discusión

Para el desarrollo de este proyecto, se trató de hacerlo lo más ajustado a las instrucciones, pero cabe resaltar que a pesar de ello, se considera que no todo salió bien. En este caso, la gramática que se utilizó para poder realizar el proceso, fue la del archivo 1.txt, esta ya no posee recursividad a la izquierda, por ello no se implementó, pensando que así serían todas las gramáticas, posiblemente, sea un error, pues es algo a considerar.

Cabe resaltar que entre los obstáculos que se presentaron en la resolución del proyecto, fueron con respecto a la implementación de chomsky, al menos, para volverlo en tal forma, pues se implementó, pero para gramáticas que posee a lo mucho, 3 caracteres, por ejemplo, X G F, si apareciera X G F A B, tronaría el programa, pues no está adaptado para tal situación. Esto se debió porque se quiso hacer un bucle indeterminado, donde terminarían, si en dado caso no hubieran derivaciones con un largo mayor a 2, pero entraba en bucle infinito, evitando tal alcance que se esperaba.

Cabe resaltar, que el proyecto estaba ya pensado, pero estaba mal planificado. Este se entregó el 15 de octubre, pero al proveerse una gramática para probar, se pudieron afrontar muchas problemáticas que poseía el mismo, esto ayudó en la resolución más precisa del mismo, pues antes no aceptaba oraciones o sentencias, sino solo palabras. Esto es algo que ayudó mucho, pero se vuelve a resaltar, que no se implementó el hecho de la recursividad por la izquierda, siendo esto algo que puede afectar en la nota final. Así mismo, se resalta que hubieron problemas para el tema de cyk, en este caso, pasándole la forma normal de chomsky, pues no reconocía sentencias. Esto afectó pues no se sabía que pasaba, y pues, acepta ciertas sentencias, pero no todas, que deberían estar en tal gramática. Esto es un factor a tomar en cuenta, pues también puede ser un indicador para la baja en pts.

Entre las recomendaciones, se puede resaltar el tema de la implementación correcta de chomsky, así como, la recursividad por la izquierda, pues con estas 2, el programa se puede volver casi inservible, ya que se depende del mismo.

El programa está compuesto por los siguiente archivos:

- **main.py**
 - *main*:
 - Método donde se ejecuta el programa.
 - *validacion_sentencias*:
 - Este método tiene como objetivo la validación de oraciones, indicando si tal oración pertenece o no en la gramática.
- **validacion_gramatica.py**
 - *validacion_gramatica*:
 - Método para validar si se ingresa una gramática válida, tanto de manera manual como una gramática directa.
- **cfg_implementacion.py**
 - *eliminar_producciones_epsilon*:
 - Elimina producciones epsilon de una gramática.
 - *eliminar_producciones_unitarias*:
 - Funcion para eliminar las producciones unitarias de la gramática.
 - *eliminar_simbolos_inutiles*:
 - Este metodo sirve para poder eliminar los elementos inútiles en la gramatica.
 - *forma_normal_chomsky*:
 - Este método lleva a la gramática a su etapa final, que es la forma normal de chomsky.
- **cyk_parse_tree.py**
 - *cyk_with_parse_tree*:
 - Implementación del algoritmo cyk.
 - *prueba_cyk*:
 - Método para realizar la prueba para verificar si una oración pertenece a la gramática.
- **cargar_gramatica.py**
 - *cargar_gramatica*:
 - Este metodo sirve para hacer la lectura de archivos donde se encuentra la gramatica

Funcionamiento del programa:

<https://youtu.be/6ZSSxPwxEDk>

Ejemplos

```

gramatica refactorizada: {'E': ['T X', 'F Y', 'A C0', ' id'], 'X': ['D C1', 'D T'], 'T': ['F Y', 'A C0', ' id'], 'Y': ['B F', 'B C3'], 'F': ['A C0', 'id'], 'A': ['('], 'B': ['*'], 'C': [')'], 'D': ['+'], 'C0': ['E C'], 'C1': ['T X'], 'C3': ['F Y']}
Ingrese una oración: id * id
id * id
gramatica refactorizada DOS: {'E': ['T X', 'F Y', 'A C0', ' id'], 'X': ['D C1', 'D T'], 'T': ['F Y', 'A C0', ' id'], 'Y': ['B F', 'B C3'], 'F': ['A C0', 'id'], 'A': ['('], 'B': ['*'], 'C': [')'], 'D': ['+'], 'C0': ['E C'], 'C1': ['T X'], 'C3': ['F Y']}
Tiempo de ejecución: 0.0 horas
$!
C3
F
 id
Y
 B
  +
  F
   id
Ingrese una oración:

```

Gramatica utilizada

$E \rightarrow T X$

$X \rightarrow + T X \mid e$

$T \rightarrow F Y$

$Y \rightarrow * F Y \mid e$

$F \rightarrow (E) \mid id$

Oraciones aceptada

- id * id
- (id * id)
-

Oraciones no aceptdad

- id + id
- (id + id)
- (+ + +)
- id +
- id *
- * id
- + id
- (id * id) + id
- (*)