Práctica 4 – gramática libre

Objetivo: Diseñar la sintaxis de cada sentencia posible en el lenguaje propio con el fin de preparar una gramática LL que permita notificar fallos durante el análisis.

Instrucciones:

1. Tomando en cuenta el programa que contiene el código de su propuesta de lenguaje, analice las variaciones que podrían tener al menos 3 tipos de sentencias y elabore una gramática siguiendo las reglas de generación de producciones, siga el ejemplo:

<PROGRAMA>

add io, common

begin

ejemplo = 20

keyboard ejemplo

display 100

ejemplo = ejemplo + 100

display ejemplo

end

<GRAMÁTICA>

*gramatica* 🡪 *encabezado listaInstr*Interfaz de usuario gráfica, Texto

Descripción generada automáticamente con confianza media

*encabezado 🡪* **add** *file* *next*

*next 🡪 , file next | , file | Ɛ*

*listaInstr 🡪* **begin** *listaInstr* **end** *|**instr listaInstr | instr | Ɛ*

*instr 🡪 id* **=** *factor |* **display** *opcion |* **keyboard** *id*

*factor 🡪 factor+factor | opcion*

*opcion🡪 id | num*

*file🡪* ***common*** *|* ***graphic*** *|* ***advance*** | ***io***

* Puede representar las sentencia usando un Autómata y posteriormente convirtiéndolo a la gramática equivalente

1. Una vez definida su gramática inicial, obtenga en secuencia los lexemas que pueden formar una sentencia y las derivaciones, por ejemplo:

<LEXEMAS> <TOKENS>

add common, io >> add file, file

add graphic >> add file

begin >> begin

display 3 >> display num

end >> end

* Use 2 ejemplos

A close-up of a graph paper

Description automatically generated

1. Identifique dos posibles errores comunes que podrían presentarse y comente la forma en que podría manejarlos en esta fase. Ejemplifique

Un error seria a la hora de asignar valores de números a un string, o que una cadena tenga números dentro o caracteres especiales.

Otro error seria al usar booleanos y que se confundan con cadenas.