**COMPARACIÓN ENTRE GRAMMAR.Y Y LA GRAMÁTICA GENERADA LALR(1)**

Al inicio, genera Los terminales y No terminales identificando los mismos gramaticalmente y en el caso del archivo .y agrega unos cuantos más de acuerdo a la programación, por ejemplo las líneas y el estado de error.

El estado 0

En el programa, es el que se genera por la gramática aumentada, el cual direcciona al estado 1, mientras que, en la gramática elaborada manualmente, contiene la gramática aumentada y todas las reglas que se derivan de ella.

El estado 1

En el programa, ejecuta las mismas acciones que el estado 0 elaborado manualmente, coincidiendo con los terminales y no terminales que deben dirigir a otros estados.

Estado 2:

En el programa, es el encargado de aceptar una palabra, con la variación que manualmente el Terminal OR debe dirigir a algún estado.

Estado 3:

La única variación es el nombramiento de los estados, pero luego coinciden en la dirección de los mismo. Manualmente es el estado I4-14

Estado 4:

Coincide con el estado I6-16 de lo generado manualmente, reduciendo ambas con la Regla de bfactor→true.

Estado 5:

Coincide con el estado I7-17 de lo generado manualmente, reduciendo ambas con la Regla de bfactor→false.

Estado 6:

En el programa, sirve para eliminar las líneas o espacios vacíos, reduciendo con una regla generada por el mismo programa.

Estado 7:

Coincide con el estado I5-15 generado manualmente.

Estado 8:

Es la coincidencia del estado I1 creado manualmente, aunque en este, se acepta la palabra en la gramática, mientras que el programa lo maneja de manera separada.

Estado 9:

Coincide con el estado I2-12 del manual, teniendo un shift y un reduce con la regla bexpr→bterm en el mismo.

Estado 10:

Coincide con el estado I3-13, en donde reduce utilizando la regla bterm→bfactor.

Estado 11:

Coincide con el estado I10-23 Que reduce con la regla bfactor→not bfactor

Estado 12:

Coincide con I11-24 del manual, desplazando únicamente a los estados correspondientes.

Estado 13:

Coincide con el estado I8-21, desplazando a los estados correspondientes.

Estado 14:

En el programa, reduce con una regla generado por el programa.

Estado 15:

Coincide con el estado I9-22, desplazando a los estados correspondientes.

Estado 16:

Coincide con el estado I20-27, reduciendo con la regla bfactor→(bexpr).

Estado 17:

Coincide con el estado I18-25, Desplazando en el terminal And y reduciendo en otros terminales con la regla bexpr→bexpr or bterm.

Estado 18:

Coincide con el estado I19-26, reduciendo con la regla bterm→bterm and bfactor.