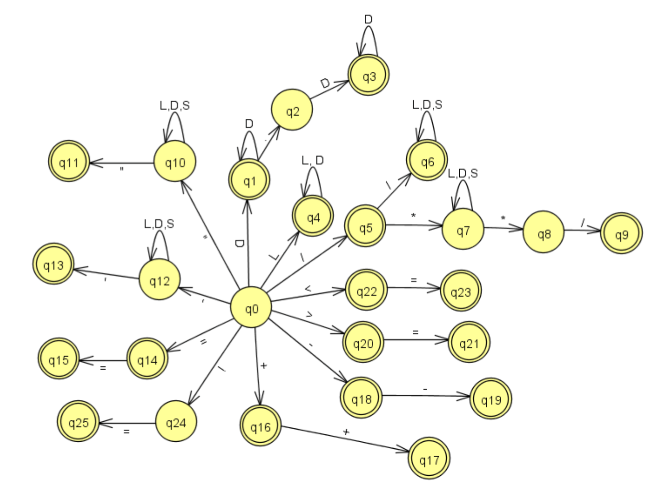
**Manual Técnico**

Dentro de este manual se detallará el proceso para determinar el AFD para el analizador léxico y la gramática libre de contexto para el análisis sintáctico.

Para el AFD se utilizó la siguiente expresión regular:

Donde D son los dígitos, L las letras y S los símbolos. En el caso de /\* (L|D|S)\* \*/ se omite el \* de los símbolos para poder saber cuando es que pasa al siguiente estado.

**D D\* | D D\*.D D\* | L (L|D|S)\* | ‘ (L|D|S)\* ’ | “ (L|D|S)\* ” | // (L|D|S)\* | /\* (L|D|S)\* \*/ | =(=|ɛ) | != | +(+|ɛ) | -(-|ɛ) | >(=|ɛ) | <(=|ɛ)**

****

Para la programación se agregaron los símbolos que faltan de token y las palabras reservadas se detectan cuando se guarda un token. Se utilizaron estados, los cuales están a continuación:

Para la gramática libre de contexto:

Ahora para determinar la gramática libre de contexto se armo viendo ejemplos redactados en el enunciado de la practica y ejemplos de entradas que pueden entrar en el proyecto.

La gramática está a continuación:

<Start> :=

WR\_Class ID S\_Open\_Key

<FunctionList>

WR\_Void WR\_Main S\_Open\_Parenthesis S\_Close\_Parenthesis S\_Open\_Key

<SentencesList>

S\_Close\_Key

<FunctionList>

S\_Close\_Key

<FunctionList> :=

<Type> ID S\_Open\_Parenthesis <Parameter> S\_Close\_Parenthesis S\_Open\_Key

<SentencesList>

<ReturnOrNot>

S\_Close\_Key <FunctionList>

<ReturnOrNot> := WR\_Return <Expression> | ɛ

<Type> := WR\_Int | WR\_Double | WR\_Char | WR\_String | WR\_Bool

<Parameter> := <ParameterDeclaration> | ɛ

<ParameterDeclaration> := <Type> ID <ParameterList>

<ParameterList> := S\_Comma <ParameterDeclaration> | ɛ

<SentencesList> :=

| <DeclarationSentence> <SentencesList>

| <AsignmentSentence> <SentencesList>

| <PrintSentence> <SentencesList>

| <IfElseSentence> <SentencesList>

| <SwitchSentence> <SentencesList>

| <ForSentence> <SentencesList>

| <WhileSentence> <SentencesList>

| <DoWhileSentence> <SentencesList>

| ɛ