

dollar = "\$"
Porcentaje = "%"
amberson = "&"
paretensisA = "("
parentesisC = ")"
asterisco = "*"
mas = "+"
coma = ","
guion = "-"
punto = "."
slash = "/"
punto = ":"
ptyComa = ";"
menor = "<"
.igual = "="
mayor = ">"
nterrogacion = "?"
arroba = "@"
corcheteA = "["
sombrero, = "^"
underScore = "_"
llaveA = "{"
or = "|"
llaveC = "}"
.colocho = "~"
slashInverso = "\\"

// Los terminales son los elementos que nos llevan a otros no terminales con los cuales podemos hacer mas producciones.

No terminales = { INICIO, CUERPO, EVALUACIONES, CONJNOTACION, LISTASIMBOLOS, SIMBOL, EXPRESION }

// El terminal con el desencadenaría todas las demás producciones es el siguiente:

Inicico = INICIO

Producciones:

// Esta es la producción que desencadena todas las demás y en ella se establece que el archivo debe empezar con una llave abierta luego se tendrá el cuerpo que se explicara posteriormente, este ira seguido de 4 signos, después de ellos vendrán las evaluaciones de las expresiones regulares que se definirían en el cuerpo.

INICIO ::= llaveA **CUERPO** porcentaje porcentaje porcentaje porcentaje
EVALUACIONES llaveC

// Esta producción nos indica que dentro del cuerpo se pueden definir conjuntos con distintas notaciones los cuales se definirán posteriormente, debido a que se pueden escribir varios conjuntos se coloco recursividad a la izquierda para evitar ambigüedad cuando únicamente se escribe un conjunto..

CUERPO ::= **CUERPO** conj dosPts identificador guion mayor **CONJNOTACION**
ptyComa

// En caso de que únicamente se desee colocar un conjunto se usara esta producción.

| conj dosPts identificador guion mayor **CONJNOTACION** ptyComa

// Dentro del cuerpo también pueden colocarse tantas expresiones regulares como se quiera por lo que se agregó recursividad por la izquierda para evitar ambigüedad cuando únicamente se escribe una expresión, debido a que una expresión puede tomar distintas formas se agrego un no terminal que definirá la forma de estas posteriormente.

| **CUERPO** identificador guion mayor **EXPRESION** ptyComa

// Esta producción se utilizara en caso de que solo se quiera colocar una expresión regular.

| identificador guion mayor **EXPRESION** ptyComa

// Como se había mencionado un conjunto puede tener dos tipos de producciones por lo que se definen en esta producción

CONJNOTACION ::= **SIMBOLO** colocho **SIMBOLO** // conjunto por rangos

| **LISTASIMBOLOS** // En caso que se quiera agregar varios elementos

// Esta producción nos sirve para listar los elementos por comas

LISTASIMBOLOS ::= **LISTASIMBOLOS** coma **SIMBOLO** // elemento que tendrá mas

| **SIMBOLO** // ultimo elemento el cual se colocara a la izquierda

// Aquí se definen las distintas formas en las que se pueden presentar dos expresiones y su operación entre ellas

EXPRESION ::= punto **EXPRESION EXPRESION** // concatenación

| or **EXPRESION EXPRESION** // conjunción

| asterisco **EXPRESION** // no viene o viene varias veces

| mas **EXPRESION** // viene una vez o mas

| interrogación **EXPRESION** // viene o no viene

| llaveA identificador llaveC // producción para un conjunto

| símbolo // en caso de que venga un símbolo cualquiera

// Estas producciones nos sirven para poder identificar todos los caracteres ASCII al momento de leer los símbolos que son miembros del conjunto.

SIMBOLO ::= admiración

| hash

| dollar

| porcentaje

| amberson

| parentesisA

| parentesisC

| asterisco
| mas
| coma
| guion
| punto
| slash
| dosPts
| ptyComa
| menor
| igual
| mayor
| interrogacion
| arroba
| corcheteA
| sombrero
| underScore
| llaveA
| or
| llaveC
| colocho
| slashInverso
| entero
| letra
| símbolo

// Por ultima esta producción permite colocar varias evaluaciones conformadas por el nombre de la expresión regular a la que pertenecen y su lexema..

EVALUACIONES ::= **EVALUACIONES** identificador dosPts simbolo ptyComa

| identificador dosPts simbolo ptyComa