

Actividad 3.2- Programando un DFA

Como está diseñando nuestro programa en Python (Tabla de Transiciones)

Diego Alberto Baños López | A01275100

José Ángel Rentería Campos | A00832436

Brenda Elena Saucedo González | A00829855

Q\Σ	[0-9]	.	E	e	=	+	-	*	/	^	[A-Z]	[a-z]	_	()	␣	\n
→ Inicio	Entero	Real1	Variable	Variable	Asignación	Suma	Resta	Multiplicación	División	Potencia	Variable	Variable	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error
Entero	Entero	Real1	Error	Error	Asignación	Suma	Resta	Multiplicación	División	Potencia	Error	Error	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Real1	Real2	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error
Real2	Real2	Error	Real3	Real3	Asignación	Suma	Resta	Multiplicación	División	Potencia	Error	Error	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Real3	Real4	Error	Error	Error	Error	Error	Real5	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error
Real4	Real4	Error	Error	Error	Asignación	Suma	Resta	Multiplicación	División	Potencia	Error	Error	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Real5	Real6	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	Error
Real6	Real6	Error	Error	Error	Asignación	Suma	Resta	Multiplicación	División	Potencia	Error	Error	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Variable	Variable	Error	Variable	Variable	Asignación	Suma	Resta	Multiplicación	División	Potencia	Variable	Variable	Variable	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Asignación	Entero	Real1	Variable	Variable	Asignación	Suma	Resta	Multiplicación	División	Potencia	Variable	Variable	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Suma	Entero	Real1	Variable	Variable	Asignación	Suma	Resta	Multiplicación	División	Potencia	Variable	Variable	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Resta	Entero	Real1	Variable	Variable	Asignación	Suma	Error	Multiplicación	División	Potencia	Variable	Variable	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Multiplicación	Entero	Real1	Variable	Variable	Asignación	Suma	Resta	Multiplicación	División	Potencia	Variable	Variable	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Potencia	Entero	Real1	Variable	Variable	Asignación	Suma	Resta	Multiplicación	División	Potencia	Variable	Variable	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
División	Entero	Real1	Variable	Variable	Asignación	Suma	Resta	Multiplicación	Comentario	Potencia	Variable	Variable	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Comentario	Fin
Paréntesis que abre	Entero	Real1	Variable	Variable	Asignación	Suma	Resta	Multiplicación	División	Potencia	Variable	Variable	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Paréntesis que cierra	Entero	Real1	Variable	Variable	Asignación	Suma	Resta	Multiplicación	División	Potencia	Variable	Variable	Error	Paréntesis que abre	Paréntesis que cierra	Inicio	Fin
Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin	Fin

En la presente tabla de transiciones se muestra los estados de transición a los que el programa de Python se mueve con cada carácter o entrada que va leyendo en un renglón del archivo de texto.

En un inicio empezamos por el estado “Inicio”, en donde va a empezar a recibir las entradas y verificar a que estado debe de moverse. Por ejemplo, al tener una entrada de tipo numérica se brinca al estado de entero, si recibe un punto “.” sabe que esta empezando a leer un dato de tipo flotante (real) por lo que se mueve al estado “Real1”, cuando recibe un carácter de tipo alfabético pasa al estado “Variable”, y cuando recibe como entrada algún operador o paréntesis, sabe que debe moverse a dichos estados. En cambio, cuando recibe como entrada un salto de línea (\n), sabe que termino de leer la expresión aritmética y pasa al estado de aceptación.

En el estado de error, una vez que se esté dentro, o que alguna transición caiga en dicho estado, ya todo las demás entradas se estarían leyendo e imprimiendo incorrectamente, ya que la palabra que se esté formando con dichas entrada da a entender, por decirlo de alguna manera, que es una palabra que NO es válida, o no está incluida en el alfabeto, por lo que cae en un estado de no aceptación.

Los estados que están en rojo dan a entender que, al momento de caer en dichos estados, ya es una expresión aritmética NO valida, pero el programa seguirá identificándolos de manera correcta, por ejemplo, teniendo esta palabra: “=/”, es una expresión aritmética no valida, pero el programa sabrá que se refiere a una asignación y a una división, por lo que no habrá problema para identificarlos e imprimirlos.

El estado “Fin”, es el estado final al terminar de leer una expresión aritmética valida, por lo que se puede considerar un estado de aceptación.