

Predicando sobre Media Types

Integrantes:

Clozza, Nicolas.

Mamone, Federico.

Tay, Lucia.

Varela Ballesteros, Catalina.

Grupo: 10

Fecha de entrega: 11/09/2018

Docente: Codagnone, Juan F.

Objetivo:

Desarrollar un programa que reciba por argumento de línea de comando un *media-range* (sección 5.3.2 Accept del RFC7231) y por entrada estándar un *media-type* (Sección 3.1.1.1 Media Type del RFC7231) por cada línea. El programa imprimirá en la salida estándar por cada *media-type* recibido los literales *true*, *false*, *null* según esté incluido por los *media-range* o la entrada esté mal formada.

Funciones destacadas de la implementación:

- `queueADT validateRange(char const *argv);`

Esta función es la encargada de validar el argumento recibido por línea de comando, así como también retornar una cola con los *media-range* obtenidos.

La implementación realizada respeta la siguiente máquina de estados:

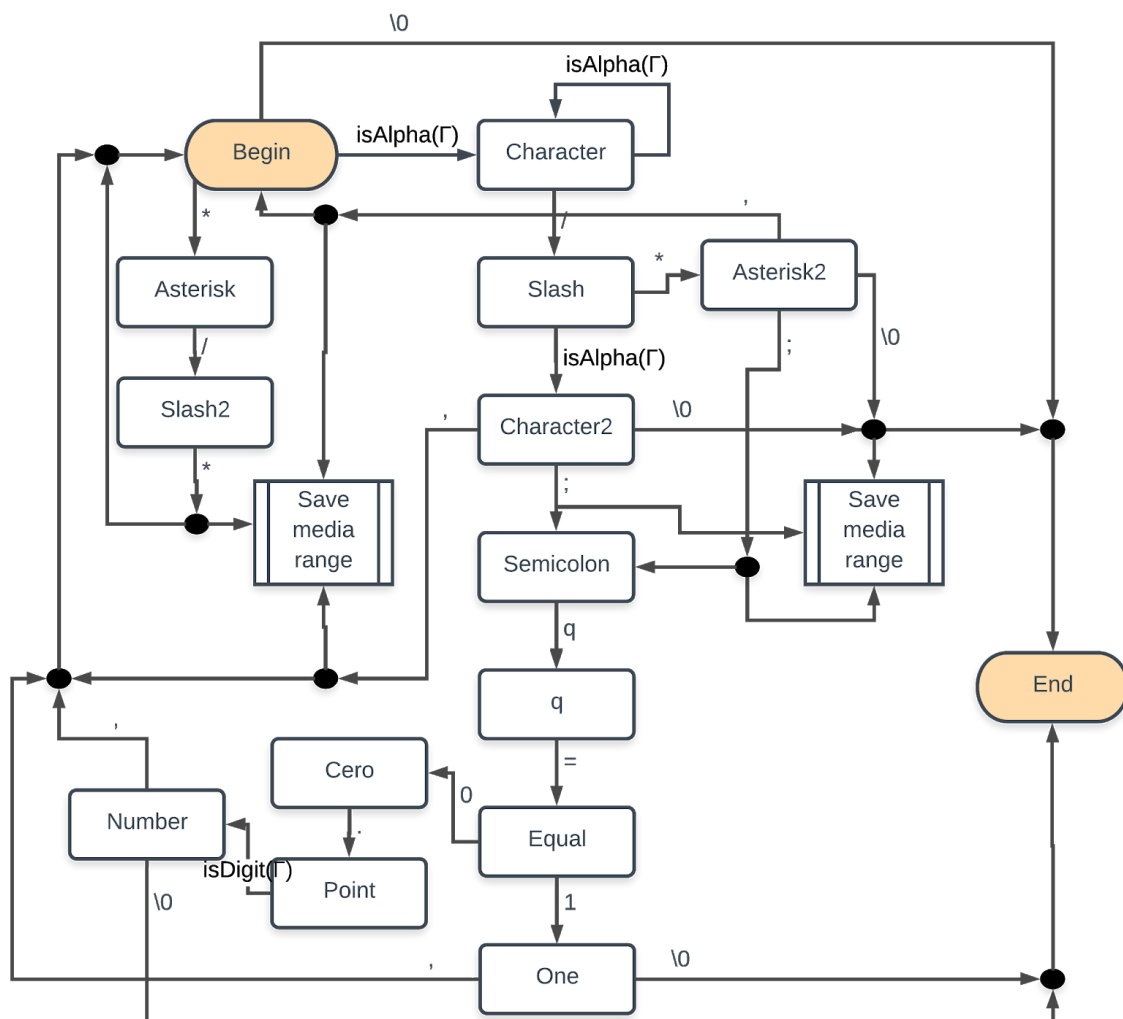


Figura 1: Máquina de estados de aceptación de los *media-range* recibidos por línea de comando.

- `queueADT validateType();`

Esta función valida los *media-type* recibidos por entrada estándar, además de almacenarlos en una cola para luego compararlos con los *media-range* previamente definidos.

La implementación realizada respeta la siguiente máquina de estados:

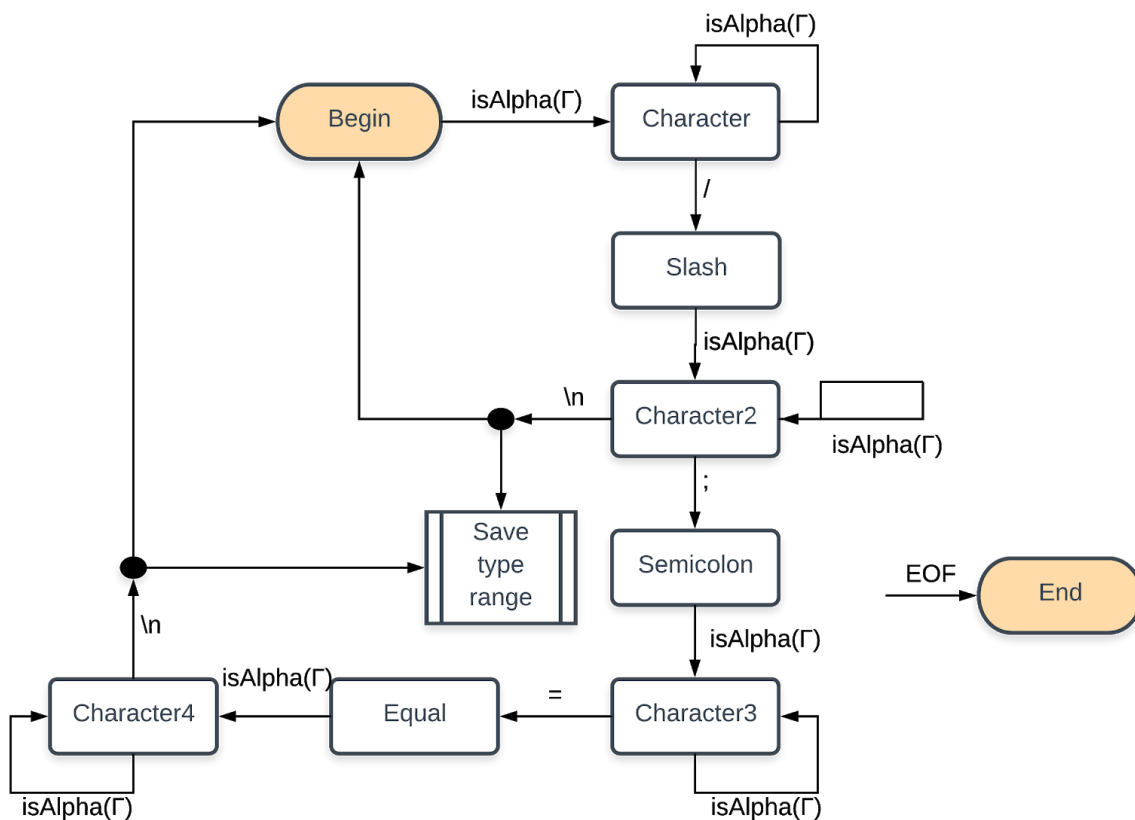


Figura 2: Máquina de estados de aceptación de los *media-type* recibidos por línea de comando.

- `void check(queueADT mediaRangeQueue, queueADT mediaTypesQueue)`

La función *check* recibe dos colas, *mediaRangeQueue* y *mediaTypesQueue*. Toma un elemento de *mediaTypesQueue* para luego iterar por sobre *mediaRangeQueue*, de esta manera puede decidir si ese elemento es aceptado o no, imprimiendo por salida estándar *true*, *false*, *null* según corresponda.

Aclaraciones:

En ambas figuras de las máquinas de estados implementadas se ignoraron todas las finalizaciones forzadas por errores.

Forma de compilación:

```
> cd protos-mediaTypes  
> make all
```

Forma de ejecución:

```
> ./match '<media-range>'
```

Finalización de la entrada estándar:

Enviar un **EOF Signal** desde el teclado. En Linux: Ctrl + D