

Informe

Descripción de las estructuras utilizadas

Brainfuck:

Recibe el nombre de los archivos de entrada y salida y el código a ejecutar. Crea un compiler y un executer que se encargan del correcto funcionamiento del mismo.

Compiler:

Recibe un código brainfuck y lo parsea para verificar la sintaxis del mismo.

Executor:

Abre los archivos de entrada y salida en caso de no ser los estándar y ejecuta el código brainfuck.

Intérprete:

Recibe el nombre de un archivo que contiene un código brainfuck. Abre y ejecuta el script en dicho lenguaje.

Producer:

Lee de entrada estándar los parámetros para la creación de branfucks. Los agrega a una cola de prioridad bloqueante, para que los n threads del pool Consumidores) los ejecuten.

CommandParcer:

Recibe un comando de tipo `(nombre_del_script, prioridad, archivo_entrada, archivo_salida, codigo_brainfuck)` y crea un brainfuck con prioridad.

BfPriority: (brainfuck con prioridad)

Es una clase contenedora. Tiene un brainfuck y la prioridad del mismo.

Consumer:

Thread que recibe una cola bloqueante, espera a que haya recursos (brainfucks) para ejecutar. Luego los ejecuta tomando siempre el más prioritario.

Esquema del diseño

Diagrama de actividad

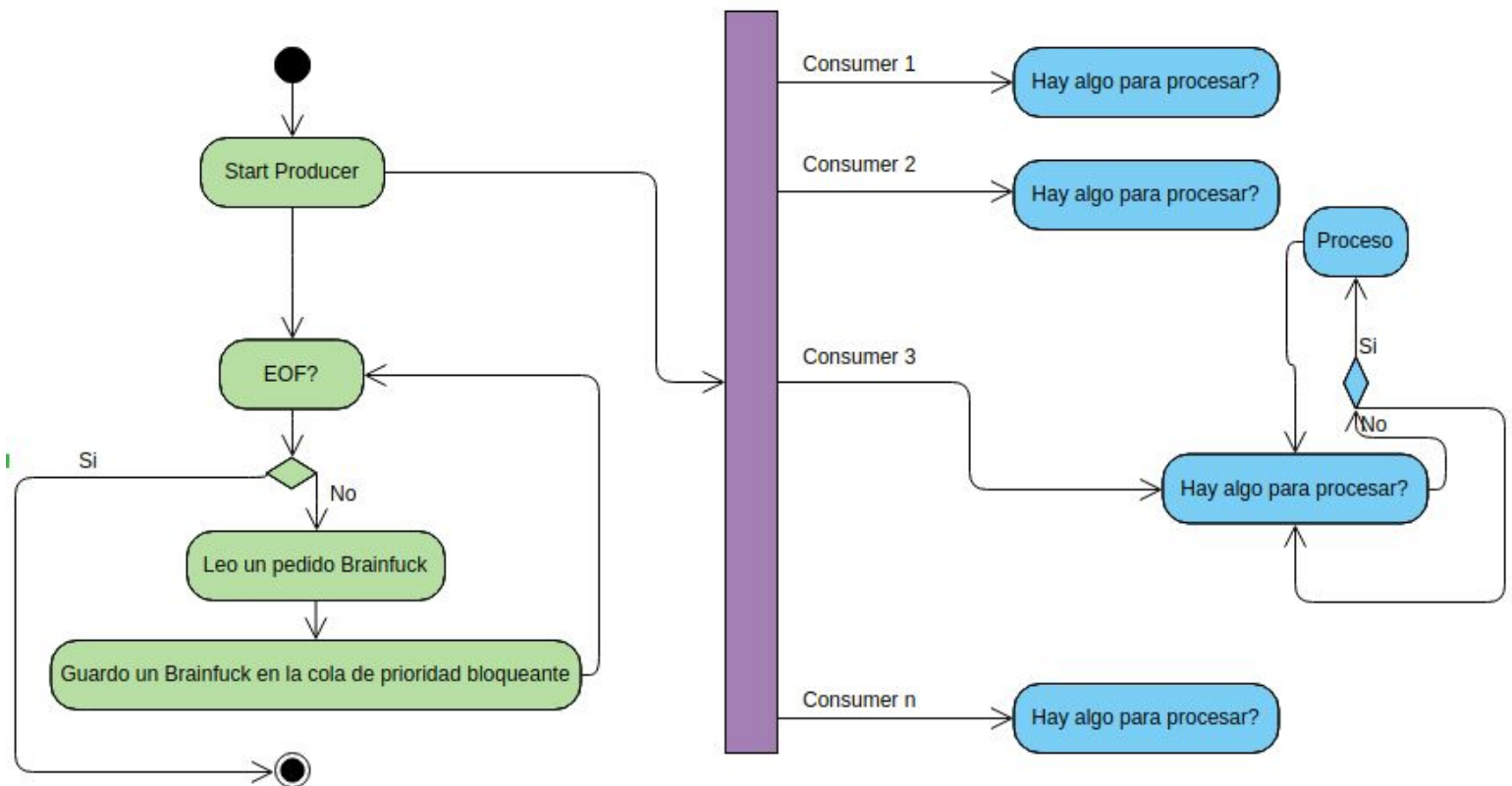


Diagrama de clases

