

Pregunta 1 [4 puntos]

Decimos que un vehículo es **VIP** si su identificador acaba en una cadena de caracteres especial (parámetro **"viptag"** -- ver abajo). Por ejemplo, si esa cadena de caracteres es **"_vip"** entonces vehículos con identificadores como **"v1_vip"** y **"car_vip"** son vehículos **VIP**.

Añade al simulador una **estrategia de cola** que avanza como máximo **"limit"** (un parámetro -- ver abajo) vehículos, dando prioridad a los vehículos **VIP** en el orden que aparecen en la cola y después al resto de vehículos siguiendo el orden de la cola. Si por ejemplo en la cola de una carretera entrante tuviéramos los vehículos **[v1_vip,c1,c2,v2_vip,c3]** en este orden de llegada y **"limit"** fuera 3, entonces saldrían los vehículos **[v1_vip,v2_vip,c1]** (en este orden). Si **"limit"** fuera 7, entonces saldrían los vehículos **[v1_vip,v2_vip,c1,c2,c3]** (en ese orden).

La entrada **JSON** que corresponde a esta estrategia tiene la siguiente sintaxis:

```
{
  "type" : "vip_dqs",
  "data" : { "viptag" : "_vip", "limit" : 5 }
}
```

donde

1. El valor de la clave **"viptag"** es la cadena de caracteres especial que identifica vehículos **VIP**, es obligatoria y, si no existe, hay que lanzar una excepción.
2. El valor de **"limit"** es el máximo número de vehículos que pueden avanzar en un tick, es opcional con valor por defecto 1.

Junto con el enunciado hay un archivo de entrada **JSON** que puedes usar para probar tu implementación (corresponde al ejemplo mencionado arriba). Puedes usar el método `s1.endsWith(s2)` de la clase `String` para comprobar si el `String s1` acaba en el `String s2`.

Para realizar esta implementación, sólo puedes añadir nuevas clases, pero no modificar ninguna otra a excepción de la clase Main. Cualquier solución que no respete este requerimiento será evaluada con 0 puntos.