

## Simulation einer Ampel zur Zufahrtsregelung

Es soll eine Ampel zur Zufahrtsregelung in Java simuliert werden.

Dazu soll auf dem Bildschirm ein geradlinig und senkrecht von unten nach oben verlaufendes Straßenstück gezeichnet werden. Die Straße soll von Autos nur einspurig in eine Richtung befahren werden können. An der Straße ist ein Haltepunkt markiert, an dem die Ampel steht.

Zur Verfügung steht eine Klasse Auto. Objekte dieser Klasse können ihre Länge angeben, sich mit Hilfe entsprechender Methoden in einem Punkt in dem Fenster positionieren, sich an ihrer aktuellen Position zeichnen, sich dort löschen, von der aktuellen Position bis zu einem anzugebenen Punkt fahren oder eine anzugebene Anzahl an Pixel in Richtung des oberen Fensterrands fahren.

Weiterhin sollen im Fenster zwei Schaltflächen mit den Aufschriften „Heranfahren“ bzw. „Weiterfahren“ zu sehen sein. Bei einem Klick auf die Schaltfläche mit der Aufschrift „Heranfahren“ soll ein neues Auto an den Haltepunkt bzw. bis an das letzte Auto, das vor dem Haltepunkt wartet, heranfahren. Die Schaltfläche mit der Aufschrift „Weiterfahren“ dient der Simulation von Grünphasen der Ampel an dem Haltepunkt. Ein Klick entspricht einer Grünphase. In jeder Grünphase darf jeweils nur ein Auto weiterfahren. Es soll von dem Haltepunkt aus die Straße entlang fahren und anschließend verschwinden. Die anderen Autos rücken nach.

Auto	
<input type="checkbox"/>	xPos: int
<input type="checkbox"/>	yPos: int
<input type="checkbox"/>	laenge: int
<input type="checkbox"/>	g: Graphics
<input type="checkbox"/>	baustelle: Baustelle
<input checked="" type="radio"/>	Auto(Baustelle, Graphics)
<input checked="" type="radio"/>	getLaenge(): int
<input checked="" type="radio"/>	setXY(int, int): void
<input checked="" type="radio"/>	warte(long): void
<input checked="" type="radio"/>	zeichneDich(): void
<input checked="" type="radio"/>	loescheDich(): void
<input checked="" type="radio"/>	fahreBis(int, int): void
<input checked="" type="radio"/>	rueckeVor(int): void

