

# Esercizi AsmetaV – Avalla

Modellazione ed Analisi di Sistemi

Università degli Studi di Milano

# Advanced Clock

## Scenario 1

- ▶ controllare che all'inizio sia mezzanotte (00:00:00);
- ▶ fare un passo di macchina;
- ▶ controllare che l'ora sia 00:00:01;
- ▶ fare un passo di macchina;
- ▶ controllare che l'ora sia 00:00:02.

## Scenario 2

- ▶ eseguire la macchina fino a quando la funzione *hours* non vale 3;
- ▶ controllare che l'ora sia 03:00:00;
- ▶ fare un passo di macchina;
- ▶ controllare che l'ora sia 03:00:01.

# Advanced Clock (2)

## Scenario 3

- ▶ impostare un invariante di scenario che afferma che l'ora è sempre minore di 3;
- ▶ impostare una par rule che, con tre update rule, modifica l'ora in 02:01:59;
- ▶ fare un passo di macchina;
- ▶ controllare che l'ora sia 02:02:00;
- ▶ impostare una par rule che, con tre update rule, modifica l'ora in 00:00:00.

## Scenario 3 – Violazione dell'invariante di scenario

Modificare lo scenario 3 per fare in modo che l'invariante di scenario sia violato.

## Scenario 1 – Prelievo corretto

- ▶ Scrivere il seguente scenario:
  - ▶ un utente inserisce la carta *card1* ed il pin corretto;
  - ▶ preleva 50 euro;
  - ▶ viene disconnesso;
- ▶ dopo ogni passo controllare che lo stato *atmState* sia corretto.

## Scenario 2 – Pin errato

Scrivere uno scenario in cui un utente inserisce il pin errato e gli viene negato l'accesso.

Controllare sempre che lo stato *atmState* sia aggiornato in modo corretto.

## Scenario 3 – Somma non disponibile sul conto

- ▶ Scrivere il seguente scenario:
  - ▶ impostare il saldo di *card1* a 10 euro;
  - ▶ autenticare *card1*;
  - ▶ provare a prelevare 50 euro;
  - ▶ successivamente provare a prelevare 10 euro.
- ▶ Dopo ogni passo controllare che lo stato *atmState* sia corretto; controllare anche che il saldo di *card1* ed i soldi disponibili nel bancomat (*moneyLeft*) siano aggiornati in modo corretto.

### Scenario 4 – Invariante di scenario riguardo la somma disponibile nel bancomat

- ▶ Impostare un invariante di scenario che afferma che la somma disponibile nel bancomat non è mai inferiore a 900 euro;
- ▶ eseguire una serie di prelievi, facendo in modo che l'invariante non sia mai violato.

# Sluice gate

## Scenario 1 – Ciclo completo apertura-chiusura

Scrivere uno scenario in cui si mostri il ciclo di vita completo della chiusa: *FULLYCLOSED* - *OPENING* - *FULLYOPEN* - *CLOSING* - *FULLYCLOSED*.

## Scenario 2 – Ciclo completo apertura-chiusura con ritardi sui passaggi di stati

Modificare lo scenario 1 per fare in modo che il passaggio da uno stato a quello successivo della chiusa avvenga in due passi di macchina.

Ogni passaggio di stato della chiusa è condizionato al valore di una determinata locazione monitorata (es. si passa da *FULLYCLOSED* a *OPENING* se *passed(closedPeriod)* è *true*): se la monitorata vale *false* la chiusa non cambia stato.

## Scenario 1

- ▶ Eseguire la macchina fino a quando  $numA = numB$ ;
- ▶ controllare che il valore di  $numA$  sia 13.

## Scenario 2

- ▶ Scrivere un invariante che afferma che  $numA$  è sempre maggiore di  $numB$ ;
- ▶ modificare i valori di  $numA$  e  $numB$ , rispettivamente, a 24 e 6;
- ▶ eseguire la macchina fino a quando  $numA = numB$ ;
- ▶ controllare che il valore di  $numA$  sia 6.



# Ferryman problem

## Scenario 1

Scrivere uno scenario in cui si mostrino tutti i passi necessari (settaggio della monitorata *carry* con il comando **set** ed esecuzione del passo con il comando **step**) a trasportare tutti gli attori sulla sponda destra.

Dopo ogni passo controllare che le posizioni degli attori siano quelle attese (comando **check**).

## Scenario 2

- ▶ Posizionare il ferryman, la goat ed il wolf sulla sponda destra;
- ▶ eseguire tutti i passi che servono per trasportare tutti gli attori sulla sponda destra.

# Coffe Vending Machine

## Scenari

Scrivere tre scenari significativi che permettano di validare il funzionamento del distributore di bevande sulla base dei requisiti del problema.

## Scenari

Scrivere tre scenari significativi che permettano di validare se il modello cattura correttamente i requisiti del gioco.