



UNIVERSITÀ
DEGLI STUDI
DI BERGAMO

Dipartimento
di Ingegneria Gestionale,
dell'Informazione e della Produzione

Asmeta e AsmetaSMV

Contatti:

Prof. Angelo Gargantini – angelo.gargantini@unibg.it
Dott. Nico Pellegrinelli – nico.pellegrinelli@unibg.it

Codice:

https://github.com/garganti/corso_unibg_tvsw/tree/master/esercitazioni/2026/1_asmeta_asmetasmv



Struttura base di una ASM in AsmetaL

```
Asm AsmName //uguale al nome del file
import StandardLibrary

signature: ...
    // Dichiarazione domini (astratti, concreti, ...)
    ...
    // Dichiarazione funzioni (monitorate, controllate, ...)
    ...

definitions:
    // Definizione di domini e funzioni
    ...
    // Definizione macro rule
    ...
    // Definizione main rule
    main rule r_Main = ...

default init initial_state:
    // Inizializzazione funzioni
    ...
```

Standard Library

- Per poter usare le funzioni e i domini base in un modello AsmetaL è necessario importare la libreria

StandardLibrary.asm



AsmetaSMV

Model checker per modelli AsmetaL



Proprietà CTL/LTL in AsmetaL

Le proprietà CTL e LTL devono essere dichiarate tra la definizione delle **macro rule** e prima della definizione della **main rule**.

- La sintassi di una proprietà CTL è:

CTLSPEC p

dove p è una espressione booleana

- La sintassi di una proprietà LTL è:

LTLSPEC p

dove p è una espressione booleana

CTL/LTL Library

- Per poter usare le funzioni CTL in un modello AsmetaL è necessario importare la libreria

CTLLibrary.asm

- Per poter usare le funzioni LTL in un modello AsmetaL è necessario importare la libreria

LTLLibrary.asm

Proprietà LTL

Connettore temporale	Significato
$X\varphi$ <i>Next</i>	è vera in s_t se e solo se φ è vera nello stato s_{t+1}
$F\varphi$ <i>Future</i>	è vera in s_t se esiste $t' > t$ tale che φ è vera nello stato $s_{t'}$
$G\varphi$ <i>Globally</i>	è vera in s_t se per ogni $t' > t$ φ è vera nello stato $s_{t'}$
$\varphi_1 \cup \varphi_2$ <i>Until</i>	è vera in s_t se esiste $t_n > t$ tale che φ_2 è vera in s_{t_n} e per ogni $t \leq t_i \leq t_{n-1}$, φ_1 è vera in s_{t_i}

Proprietà CTL

Connettore temporale	
A Along all paths -> INEVITABILMENTE	X
	F
	G
	U
E Along at least (there exists) one path -> POTREBBE SUCCEDERE	X
	F
	G
	U

NuSMV

- **AsmetaSMV** sfrutta il model checker NuSMV che può essere scaricato da:
<http://nusmv.fbk.eu/>
- Estrai i file
- Aggiungi NuSMV/bin nel path
- Riavvia il PC



Sistema di Autenticazione



Sistema di Autenticazione - Modellazione

Si vuole modellare un sistema di autenticazione che gestisce un insieme finito di utenti.

Per ogni utente il sistema mantiene uno stato dell'account e un numero di tentativi di autenticazione falliti. Ogni account può trovarsi in uno dei seguenti stati:

- **ATTIVO**, quando l'utente può ancora tentare l'autenticazione;
- **AUTENTICATO**, quando l'utente ha effettuato con successo il login;
- **BLOCCATO**, quando non sono più consentiti ulteriori tentativi di autenticazione.

Quando viene ricevuta una richiesta di autenticazione, vengono forniti l'utente richiedente e la password inserita.

Esiste un numero massimo di tentativi falliti consentiti, pari a 3. Se il numero di tentativi falliti è minore del massimo consentito, e la password inserita coincide con la password associata a quell'utente, l'utente passa nello stato AUTENTICATO; altrimenti il numero di tentativi falliti di quell'utente viene incrementato di uno. Una volta AUTENTICATO, un utente non può più cambiare stato.

Se invece il numero di tentativi falliti ha già raggiunto il valore massimo consentito, allora l'utente passa nello stato BLOCCATO.

Si assumono, per semplicità, tre utenti del sistema, indicati come utente1, utente2 e utente3. A ciascuno di essi è associata una password corretta prefissata. All'inizio, per ogni utente, lo stato è ATTIVO e il numero di tentativi falliti è 0.

Sistema di Autenticazione - Verifica

Si modellino in CTL e/o LTL le seguenti proprietà del sistema di autenticazione:

1. Esiste uno stato in cui tutti gli utenti sono autenticati
2. Se un utente è bloccato, allora il suo numero di tentativi è pari al massimo
3. Se un utente è bloccato, allora rimane sempre bloccato in futuro
4. Se utente1 ha raggiunto il numero massimo di tentativi e richiede autenticazione, allora viene bloccato
5. Se utente1 non ha esaurito i tentativi disponibili ed effettua una richiesta di autenticazione corretta, allora nel passo successivo è autenticato e rimane sempre autenticato
6. Per ogni utente, il numero di tentativi è sempre compreso tra 0 e il numero massimo di tentativi
7. Se un utente è bloccato, allora non può più passare allo stato autenticato
8. utente1 rimane attivo finché non si autentica oppure viene bloccato

Successivamente, si utilizzi AsmetaSMV per verificare che le proprietà formalizzate risultino soddisfatte dal modello.

Infine, si definisca almeno una proprietà volutamente errata, e si verifichi che AsmetaSMV produca un controesempio.



Ferryman

Ferryman - Modellazione

Un ferryman deve portare sull'altra sponda di un fiume un wolf, una goat ed un cabbage, e può trasportarne solo uno per volta.

Ci sono due situazioni di pericolo:

- Il wolf mangia la goat se il ferryman non è presente a controllare;
- La goat mangia il cabbage se il ferryman non è presente a controllare;

All'inizio sono tutti sulla sponda LEFT. Ad ogni passo, l'utente decide chi deve essere trasportato sull'altra sponda dal FERRYMAN: la GOAT, il CABBAGE, il WOLF, oppure fare attraversare il fiume con nessuno a bordo (il FERRYMAN trasporta solo se stesso).

Variante: Modificare la specifica in modo che la scelta venga effettuata casualmente ma non porti mai ad una situazione di pericolo.

Ferryman - Verifica

Scrivere le seguenti proprietà:

1. Se su una sponda ci sono goat e cabbage, deve esserci anche il ferryman
2. Se su una sponda ci sono wolf e goat ci deve essere anche ferryman

Verificare sulla versione con input dall'utente.

Scrivere le seguenti proprietà:

3. Esiste uno stato in cui tutti si trovano sulla sponda di destra
4. E' possibile che goat sia da solo sulla sponda di sinistra
5. E' possibile che il cabbage sia da solo sulla sponda di destra

Verificarle tutte le proprietà (anche le prime due) sulla variante automatica.

Verificare la negazione della proprietà 3: ci da la soluzione al rompicapo!