

Exercícios

Verificação de Modelos NuSMV/NuXMV + Fasten

1) Seja o seguinte código para o verificador de modelos NuSMV/NuXMV/Fasten:

```
MODULE main
VAR
    a : boolean;
    b : boolean;
ASSIGN
    init(a) := FALSE;
    init(b) := FALSE;
    next(a) := case
        !a : TRUE;
        b : TRUE;
        TRUE : {FALSE, TRUE};
    esac;
    next(b) := case
        !b : {FALSE, TRUE};
        TRUE : TRUE;
    esac;
```

Com base no código, responda as seguintes questões:

a) Apresente um Modelo de Kripke correspondente ao sistema de transição de estados codificado.

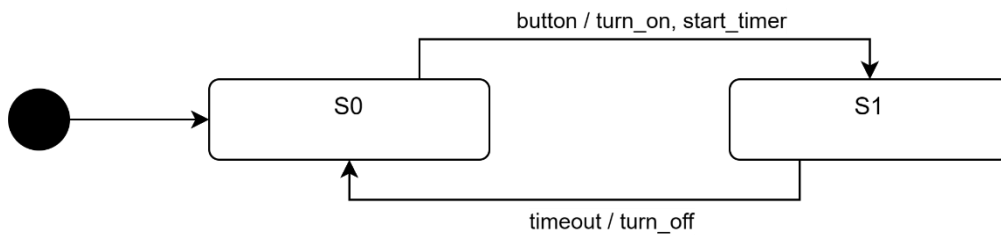
b) Para cada fórmula LTL a seguir, verifique através do NuSMV/NuXMV/Fasten se ela é verdadeira ou não. Caso seja falsa, mostre um caminho no Modelo de Kripke que seja um contraexemplo.

- $a \wedge b$
- Xa
- $F(\neg a \wedge b)$
- GFa
- $GF\neg a$
- FGa
- $G(b \Rightarrow Fa)$
- $\neg a Ub$
- $(GFa) \Rightarrow (GFb)$

c) Para cada fórmula CTL a seguir, verifique através do NuSMV/NuXMV/Fasten se ela é verdadeira ou não. Caso seja falsa, mostre um contraexemplo.

- $EG(\neg b)$
- AFa
- $EF AG(a \wedge b)$
- EGa
- $AG(a \vee b)$

2) Seja o seguinte diagrama de estados da UML (<https://www.baeldung.com/cs/uml-state-diagrams>) para o controle de uma lâmpada com temporizador para desligamento:



Para o modelo, assumir que:

- *timer* é uma contagem decrescente no intervalo de 5 (valor inicial) a 0 (valor de *timeout*);
- *button* corresponde a um evento externo ao sistema de botão pressionado;
- *timeout* corresponde a um evento de término da contagem do *timer*;
- *turn_on* é uma ação que corresponde a ligar uma lâmpada;
- *turn_off* é uma ação que corresponde a desligar uma lâmpada;
- *start_timer* é uma ação que corresponde a iniciar a contagem do *timer*.

a) Complete o seguinte código de acordo com as informações da modelagem.

```

MODULE main
IVAR
    button: boolean;

VAR
    timer: 0..5;
    estado: {s0, s1};

ASSIGN
    init(estado) := s0;

    next(estado) := case
        _____
        esac;

    next(timer) := case
        _____
        esac;
  
```

DEFINE

 lampada := estado = s1;

 timeout := timer = 0;

b) Defina fórmulas temporais em LTL para as seguintes propriedades:

- A lâmpada sempre irá ligar se o botão for pressionado.
- Uma vez que a lâmpada esteja ligada, se não for pressionado o botão, ela não voltará a estar ligada.

c) Verifique se as propriedades valem para seu modelo.

3) Retome o exercício do Modelo de Kripke para a representação de uma lâmpada que pode ser ligada/desligada e que também pode queimar.

a) Codifique um modelo como um arquivo NuSMV/NuXMV.

b) Através da ferramenta Fasten, edite o model através do diagrama de estados e compare o código gerado para o NuSMV/NuXMV a partir do diagrama de estados com o código criado no exercício “a”.

c) Verifique através de fórmulas LTL ou CTL, utilizando o NuSMV/NuXMV/Fasten, as seguintes propriedades:

- Uma lâmpada pode estar ligada no futuro.
- Uma lâmpada eventualmente pode permanecer sempre queimada.
- Uma lâmpada não pode passar de desligada para queimada.
- Uma lâmpada pode voltar a estar ligada depois de queimada.