

Nome: Carolina Afonso

1. Considera o sistema de transições $T = (S = \{s_0, s_1, s_2, s_3\}, \{s_0 \rightarrow s_2, s_0 \rightarrow s_1, s_1 \rightarrow s_1, s_1 \rightarrow s_2, s_1 \rightarrow s_3, s_2 \rightarrow s_0, s_2 \rightarrow s_1, s_2 \rightarrow s_2, s_3 \rightarrow s_0, s_3 \rightarrow s_3\}, \{s_0\}, L(s_0) = \{x_1, x_2\}, L(s_1) = \{x_1\}, L(s_2) = \{\}, L(s_3) = \{x_2\})$.
 - (a) Desenha o sistema de transições T .
 - (b) Usando o algoritmo de etiquetagem diz se $T \models E(x_2 \cup x_1)$.
 - (c) Utilizando a ordem $[x_1, x_2]$, determina OBDD's para representar o conjuntos de estados $\{s_0, s_2\}$.
 - (d) Determina a tabela de verdade para a relação de transição utilizando a ordem $[x_1, x_2, x'_1, x'_2]$.
 - (e) Desenha o OBDD para a relação de transição (utilizando a ordem da alínea anterior).
 - (f) Aplica o algoritmo de etiquetagem (adaptado à representação por OBDD's e utilizando a ordem $[x_1, x_2]$) ao modelo T , para determinar os conjuntos de estados onde se verifica a fórmula $E(x_2 \cup x_1)$.
2. Usando a função $F(X) = Sat(\varphi) \cap pre_{\forall}(X)$ demonstra que $Sat(AG\varphi)$ é o ponto fixo máximo de F .
3. (Semáforos de transito)

Considera dois semáforos com três luzes (vermelho, verde e amarelo) (podendo também estar desligados). A sequencia de cores deve ser a habitual: vermelho, verde, amarelo.

- (a) Escreve um programa para o Spin que implemente os dois semáforos e que inicialmente estejam os dois com luzes diferentes.
- (b) Testa o funcionamento executando o comando `spin -u<numero> <nome>.pml` (onde número é o numero de passos) ou


```
% spin -a <nome>.pml
% gcc -o pan pan.c
% ./pan
```

 No comando `pan` podes quer usar as opções `-a` ou `-d`, p.e. No spin a opção `-t` permite ver um caminho em que tenha ocorrido um erro: `spin -t -p <ficheiroerro>`
- (c) Escreve fórmulas do LTL que correspondam às seguintes propriedades:
 - i. Os semáforos nunca estão com uma luz da mesma cor simultaneamente.
 - ii. O semáforo 1 está um número infinito de vezes com a cor verde.
- (d) Para testar as fórmulas podes escrevê-las num ficheiro ou na linha de comando considerando que:

```
F → < >
G → []
X → X
U → U
∧ → &&
∨ → ||
¬ → !
→ → - >
```

Por exemplo GFp corresponde a `[]<>p`. Na linha de comando do "spin" usar a opção `-f` p.e.: `spin -a -f '[]<>p' nome.pml`. Ou seja:

```
% spin -a -f <formula> <nome>.pml
% gcc -o pan pan.c
% ./pan -a -f
```

Se escreveres a fórmula num ficheiro (p.e `forl1.prp`):

```
% spin -a -F forl1.prp <nome>.pml
```

```
% gcc -o pan pan.c
```

```
% ./pan -a -f
```

ou

```
% spin -a -F forl1.prp > forl1.pml
```

```
% spin -a -N forl1.pml <nome>.pml
```

```
% gcc -o pan pan.c
```

```
% ./pan -a -f
```

Neste último caso, a fórmula foi transformada num programa Promela `forl1.pml` com a instrução `never`.