Detalhes dos Programas

**1 – OrdenadorEstratégia**

Esse programa consiste em organizar a lista de estratégia a partir de uma variável a escolha. Meus modelos usam uma variável “w” que é incrementada a cada transição com evento controlável, porem a lista gerada vem em uma ordem aleatória, sendo assim criei o programa para organizar essa lista.

Algoritmo: O algoritmo varre toda linha da lista que comece com a palavra “State”, e ao encontrar a variável de contagem de eventos, no nosso caso o “w”, ele para de varrer e armazena em um vetor secundário a linha do “state”, a linha a baixo começando com “When you are in” e a última linha, até que ele encontre uma linha com string vazia, ou seja, uma linha em branco, eliminando então essas linhas da lista. O algoritmo faz isso com todos os estados e quando ele chega ao final da lista e não há mais linhas começando com a palavra “State”, ele finaliza a busca e gera uma nova lista com os estados organizados.

**2 – EditaArquivo**

Esse programa é usado para eliminar certas variáveis que são sempre incrementadas, como por exemplo a variável “w”, que está sempre crescendo à medida que o sistema funciona. Outras variáveis que também são retiradas são: a de produção, “Production”, e as variáveis de insumo como “Base” e “Cover”, que são variáveis que estão sempre decrementando a medida que elas são solicitadas no ASRS.

Algoritmo: O programa, ao receber o nome das variáveis que precisam ser excluídas, varre a lista atrás destas. Indo na mesma ordem, ela procura essas variáveis levando em conta uma string de um slot vazia na frente da variável e um sinal de igual, “=”, após a variável, assim o programa reconhece que essa variável está no local certo, na linha do estado, para ser excluída.

**3 – IgualdadeArquivo**

Esse programa foi o que agilizou o ganho de tempo do trabalho e seguiu uma ideia que tive, que era de verificar quando o sistema alcança o mesmo estado dentro da sequência de eventos. Inicialmente pensei em faze-lo verificar tanto os locais quanto os relógios, porém, ao verificar os relógios, ele encontra um problema que é a variável “Global” dentro do conjunto de relógios. Essa variável não foi retirada como havia sido feito no processo anterior com as variáveis de estado, pois existem relações de guarda em que toda ela envolve o relógio “Global”, ou seja, se eu retiro esse relógio, o guarda não representa mais nada, assim sendo permaneci com este relógio e olhei apenas para os locais.

Algoritmo: O programa varre a lista atrás das linhas que começam com a palavra “State”, e armazena toda a linha em um vetor secundário. Ele começa pela primeira linha e verifica se existe alguma linha igual a essa, caso sim, ele escreve na frente da linha armazenada e da encontrada, a descrição “Igualdade – x,y;” sendo x a linha armazenada e y a linha igual a x. O programa repete essa execução para todas as linhas, excluindo as que já foram armazenada, até que a lista acabe, gerando então uma nova lista com os estados, que possuem os locais iguais, marcados no início da linha.

**4 – XMLGenerator**

Esse programa elimina um grande trabalho, que é a criação do automato do controlador. Ele gera um arquivo XML já com o controlador pronto, porem ainda não está fechando os ciclos, essa tarefa estou fazendo manualmente por enquanto. Creio que ainda esse mês ele já esteja fechando os ciclos também.

Algoritmo: O programa pega a lista, depois de ter passado pelos outros programas, e faz um scan de quantos “States” existem. Após esse scan, ele gera a primeira parte do arquivo xml, que é a sequência de locais, posição dos locais e o nome deles. Exemplo, digamos que a lista tinha 3 linhas começando com “State”, então esse programa vai gerar a primeira parte do código em xml da seguinte maneira:

<?xml version="1.0"?>

[<nta>](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[<template>](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

<name>Controller</name>

[<location y="**0**" x="**0**" id="**id0**">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

<name y="**-8**" x="**16**">S0</name>

</location>

[<location y="**64**" x="**0**" id="**id1**">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

<name y="**56**" x="**16**">S1</name>

</location>

[<location y="**128**" x="**0**" id="**id2**">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

<name y="**120**" x="**16**">S2</name>

</location>

[<location y="**192**" x="**0**" id="**id3**">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

<name y="**184**" x="**16**">S3</name>

</location>

<init ref="**id0**"/>

Ou seja, para 100 elementos, ele vai criar 100 locais, com suas coordenadas, seu nome e seu “id”, colocando ainda o “id0” como o estado inicial. (com base nesse “id” que farei mais tarde o programa fechar os ciclos sozinho).

As coordenadas seguem um padrão que criei, tendo uma sequencia de 13 locais na vertical, após o 13 ele anda x para a direita e criar mais 13 na vertical novamente, e assim vai até acabar os elementos da lista.

A segunda parte ele cria as transições analisando a linha que começa com “When you are in”. Ele procura onde está os elementos usados como guardas, no caso os relógios que estão sempre entre “()”, além de procurar o evento que esta ocorrendo que se encontra na mesma linha na lista., assim ele gera o seguinte código.

[<transition action="">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

<source ref="**id0**"/>

<target ref="**id1**"/>

[<label y="**23**" x="**-64**" kind="**guard**">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[<![CDATA[](file:///C:\\Users\\eocc\\Documents\\IME\\Dissertação\\Modelos\\Modelo_ESTACAO_6_MECATRIME\\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[Global==0](file:///C:\\Users\\eocc\\Documents\\IME\\Dissertação\\Modelos\\Modelo_ESTACAO_6_MECATRIME\\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[]]>](file:///C:\\Users\\eocc\\Documents\\IME\\Dissertação\\Modelos\\Modelo_ESTACAO_6_MECATRIME\\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

</label>

<label y="**36**" x="**-64**"kind="**synchronisation**">d0!</label>

</transition>

[<transition action="">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

<source ref="**id1**"/>

<target ref="**id2**"/>

[<label y="**87**" x="**-64**" kind="**guard**">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[<![CDATA[](file:///C:\\Users\\eocc\\Documents\\IME\\Dissertação\\Modelos\\Modelo_ESTACAO_6_MECATRIME\\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[sc1==88 && sc1==Global && Global==88](file:///C:\\Users\\eocc\\Documents\\IME\\Dissertação\\Modelos\\Modelo_ESTACAO_6_MECATRIME\\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[]]>](file:///C:\\Users\\eocc\\Documents\\IME\\Dissertação\\Modelos\\Modelo_ESTACAO_6_MECATRIME\\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

</label>

<label y="**100**" x="**-64**" kind="**synchronisation**">a0!</label>

</transition>

[<transition action="">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

<source ref="**id2**"/>

<target ref="**id3**"/>

[<label y="**151**" x="**-64**" kind="**guard**">](file:///C:\Users\eocc\Documents\IME\Dissertação\Modelos\Modelo_ESTACAO_6_MECATRIME\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[<![CDATA[](file:///C:\\Users\\eocc\\Documents\\IME\\Dissertação\\Modelos\\Modelo_ESTACAO_6_MECATRIME\\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[ss1-Global==-88 && Global==88](file:///C:\\Users\\eocc\\Documents\\IME\\Dissertação\\Modelos\\Modelo_ESTACAO_6_MECATRIME\\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

[]]>](file:///C:\\Users\\eocc\\Documents\\IME\\Dissertação\\Modelos\\Modelo_ESTACAO_6_MECATRIME\\Lista_8_Producoes2_ordenado_editado_igualdade_controller.xml)

</label><label y="**164**" x="**-64**" kind="**synchronisation**"> d1!</label>

</transition>

Ou seja, ele diz de onde a transição está saindo, para onde ela esta indo, as coordenadas da escrita “guard”, o conteúdo do “guard”, as coordenadas da escrita “synchronisation” e o conteúdo de “synchronisation”, no caso o evento. Ao fazer isso com a lista toda, ele finaliza o programa gerando o arquivo xml completo, pronto para ser importado pelo software UPPAAL TIGA.

Todas as coordenadas desse programa foram pré-estabelecidas por mim, após criar um modelo padrão dentro do UPPAAL.

**Como usar os programas para gerar o controlador**

**Toodos os programas precisam estar na mesma pasta da lista**

1. Abra o primeiro programa (Ordenação), e coloque o nome da lista dentro do programa, ele irá pedir o nome da variável de ordenação, basta digitar e clicar em ok, nesse caso é a variável “w”. Após isso ele vai gerar uma nova lista ordenada com o nome modificado com “Lista\_ordenado” no final.
2. Copie o nome da nova lista “Lista\_ordenado” e abra o segundo programa (edição), e coloca o nome da lista nele, clique em ok. Agora ele irá pedir para por as variáveis que vc deseja deletar, detalhe, ele só vai deletar as variáveis dentro da linha do “State”. Coloque as variaves, que nesse caso são: “Production;w;Base;Cover”, sempre com ponto e virgula entre elas, e clique em ok. O programa vai gerar uma nova lista com o nome “Lista\_ordenado\_editado”.
3. Agora com a nova lista editada, copie o nome e abra o 3 programa (igualdade), e basta por o nome da lista no campo e clicar em ok que ele irá gerar uma nova lista com o nome “Lista\_ordenado\_editado\_ repeatedStates”. Nessa nova lista, o programa listou os estados que são iguais.
4. Com a nova lista, basta abrir o quarto programa (Gerador do Controlador em xml), e colocar o nome da nova lista, clicando em ok depois. Esse programa irá criar o arquivo xml com o controlador, “Lista\_ordenado\_editado\_ repeatedStates\_controller”. Basta agora ir no UPPAAL e com seus modelos já carregados, mandar importar o automato que esta dentro desse xml

Falta fazer o programa achar sozinho os ciclos, e no final de tudo, farei um único programa que fara tudo isso de uma só vez, colocando apenas a primeira lista gerada pelo UPPAAL, ele vai entregar o controlador.