

Relatório programa de simulação

Este relatório consiste em descrever e explicar o funcionamento do programa implementado para a disciplina Modelagem e Simulação. Este relatório não descreve o código-fonte do programa, pois como dizia no enunciado, o usuário não conhece nenhuma linguagem de programação. O programa trabalha apenas com números inteiros (exceto para as estatísticas) pois desta forma é mais fácil verificar o funcionamento. O simulador foi feito em Java.

Classes

O programa contém 16 classes no total, sendo elas:

- Chegada – Classe que implementa um evento do tipo chegada.
- Entidade – Classe que define os atributos das entidades
- EstadoServidor – Enumerador que define os possíveis estados do servidor.
- Estatistica – Singleton usado para calcular as estatísticas do trabalho
- Evento – Classe abstrata que define um evento.
- Falha – Classe que implementa um evento de falha no servidor.
- FalhaConsertada – Classe que implementa um evento de conserto do servidor.
- Gerador – Classe utilizada para gerenciar e gerar eventos.
- ListaEncadeadaOrdenada – Classe que implementa a lista de eventos futuros.
- Relogio – Classe que implementa o tempo atual da simulação
- Saida – Classe que implementa um evento do tipo saída
- Simulador – Classe principal do programa.
- Tipo – Enumerador para definir o tipo de entidade
- Init – Tela inicial do programa
- TelaDeExecucao – Tela de execução do programa.

Rotinas

O programa é executado com base no algoritmo de eventos futuros, ou seja, a simulação avança de acordo com o evento com o tempo mais baixo. Primeiramente o programa recolhe as informações de entrada fornecidas pelo usuário. Em seguida, inicializa as variáveis do programa no método setup() e então entra no loop que executa os eventos que existem na lista de eventos futuros.

No setup() é gerado um evento chegada inicial (dependendo da porcentagem escolhida pelo usuário) e também é gerado a falha dos dois servidores. Sempre que uma entidade chega, esta gera um evento de chegada para outra entidade (1 ou 2) e estando o servidor livre, ocupando gerando um evento de saída. Caso o servidor esteja ocupado, entra na fila do servidor e caso esteja em falha, troca de servidor. Sempre que um evento de saída é executado e existem entidades na fila, ele gera outro evento de saída. Sempre que um evento de falha é executado,

ele gera o evento que tornará o servidor livre novamente. Caso exista uma entidade no servidor e ele entra em falha, o tempo de saída dela é ajustado para um novo evento de saída. Sempre que o evento conserta o servidor, ele gera o próximo evento em que o servidor falhará novamente (tempo entre falhas).

O programa termina quando ele chega no seu limite do tempo de execução informado, ou seja, pode haver entidades que entraram mas não saíram. Estas entidades (quantidade) são mostradas nas estatísticas finais e não são usadas para calcular outras estatísticas.

Distribuições

Este programa possui 5 tipos de distriuições para o usuario escolher: Normal, Uniforme, Triangular, Exponencial e Constante. Estas distribuições foram implementadas com o auxilio da biblioteca “commons-math-3-3.3.jar”. O usuário pode escolher entre estas 5 distribuições para qualquer valor de entrada desejado.

Estatísticas

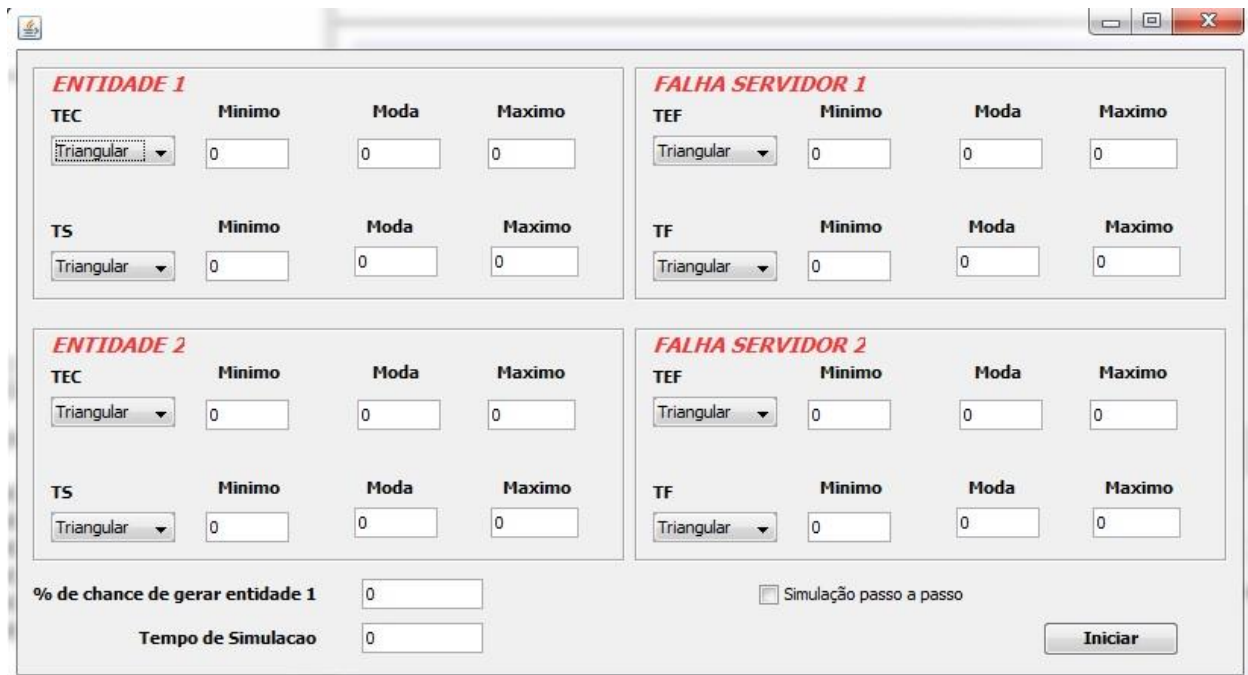
Ao final da simulação, é mostra as estatisticas baseadas nas entradas fornecidas pelo usuario. São elas:

- Entidades no sistema – Quantidade de entidades atualmente no sistema
- Trocas da entidade 1 – Quantidade em que a entidade tipo 1 trocou de servidor por motivos de falha
- Tempo médio na fila entidade 1 – Ttempo médio que uma entidade tipo 1 permaneceu na fila
- Tempo em falha servidor 1 – Tempo total em que o servidor 1 permaneceu em falha
- Tempo médio no sistema entidade 1 – Tempo médio em que a entidade 1 permaneceu no sistema
- Tempo em falha servidor 2 - Tempo total em que o servidor 2 permaneceu em falha
- Quantidades de entidades já passaram – Quantidade de entidades (1 ou 2) que já passaram pelo programa.
- Trocas entidades 2 – Quantidade em que a entidade tipo 2 trocou de servidor por motivos de falha
- Tempo medio na fila entidade 2 – Tempo medio em que a entidade 2 permaneceu na fila
- Tempo medio no sistema entidade 2 – Tempo medio em que a entidade 2 permaneceu no sistema
- Quantidade de falhas servidor 1 – Numero de vezes em que o servidor 1 falhou
- Quantidade de falhas servidor 2 – Numero de vezes em que o servidor 2 falhou

Obs: As estastísticas de tempo na fila só consideram as entidades que ocuparam a fila, ou seja, a media é feita na quantidade de entidades que tiveram que ir para a fila.

Utilizando o programa

Inicialmente, é preciso ter a última versão do Java instalado no computador para executar o programa, em seguida dê dois cliques no arquivo “INE5425.jar” e então aparecerá a tela inicial do programa:



A tela inicial do programa apresenta uma interface com quatro painéis de configuração para entidades e falhas de servidor, além de controles gerais de simulação.

ENTIDADE 1			
TEC	Minimo	Moda	Maximo
Triangular	0	0	0
TS	Minimo	Moda	Maximo
Triangular	0	0	0

FALHA SERVIDOR 1			
TEF	Minimo	Moda	Maximo
Triangular	0	0	0
TF	Minimo	Moda	Maximo
Triangular	0	0	0

ENTIDADE 2			
TEC	Minimo	Moda	Maximo
Triangular	0	0	0
TS	Minimo	Moda	Maximo
Triangular	0	0	0

FALHA SERVIDOR 2			
TEF	Minimo	Moda	Maximo
Triangular	0	0	0
TF	Minimo	Moda	Maximo
Triangular	0	0	0

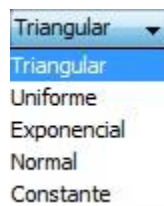
% de chance de gerar entidade 1: 0

Tempo de Simulacao: 0

☐ Simulação passo a passo

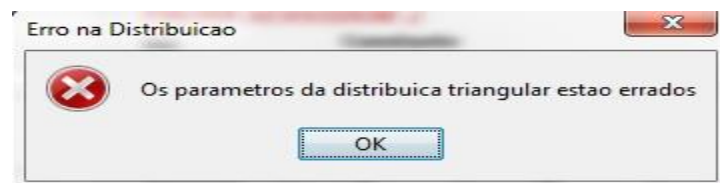
Iniciar

Nesta tela o usuário pode definir se deseja executar uma simulação passo a passo ou executar direto (padrão), também é possível escolher as distribuições para cada entrada possível. Estas são as opções:

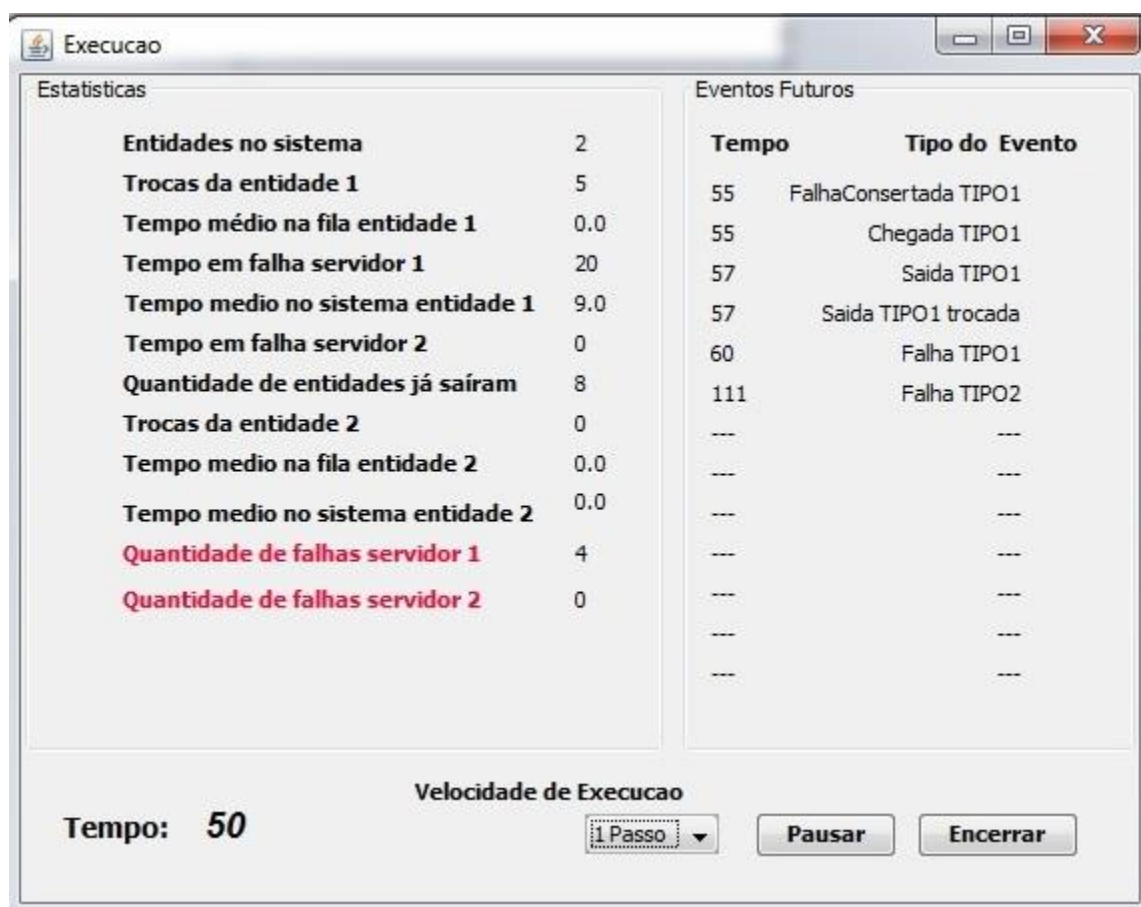


Obs: o tempo em falha não deve ser menor que o tempo entre falhas, portanto é recomendado utilizar uma distribuição constante.

Caso o usuário insira algum valor inadequado para alguma distribuição, uma mensagem de erro é mostrada:

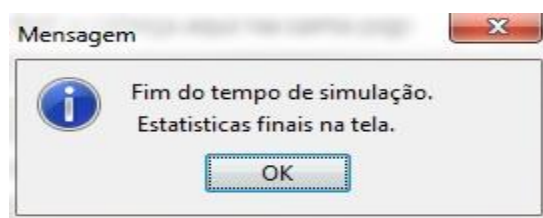


Depois de configurado tudo, clique em Iniciar e o programa mudará para a tela de execução:



Nesta tela, é possível acompanhar o andamento do programa conforme o avanço do tempo. Do lado direito da tela, temos a lista de eventos futuros que mostra o evento e o tipo de entidade. Também mostra as saídas cuja entidade teve que trocar de servidor (para diferenciar das saídas padrão). Do lado esquerdo, as estatísticas variando conforme o evento.

Após passar todo o tempo de execução, uma mensagem avisando o fim da simulação é mostrada e as estatísticas finais aparecem na mesma tela.



Para encerrar o programa, basta clicar no "X" do canto superior direito.

Para executar o programa com outros parâmetros, basta executar novamente o arquivo "INE5425.jar".