

UM SISTEMA ANALISADOR CONVERSACIONAL
PARA SIMULAÇÃO

VALDEMAR RODRIGUES FILHO

UM SISTEMA ANALISADOR CONVERSACIONAL PARA SIMULAÇÃO

Valdemar Rodrigues Filho

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS (M.Sc.).

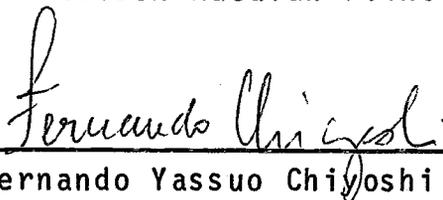
Aprovada por:



Ricardo Spinelli de Carvalho
Ricardo Spinelli de Carvalho
(Presidente)



Nelson Maculan Filho
Nelson Maculan Filho



Fernando Yassuo Chiyoshi
Fernando Yassuo Chiyoshi

RIO DE JANEIRO, RJ - BRASIL
ABRIL DE 1981

RODRIGUES FILHO, VALDEMAR

Um Sistema Analisador Conversacional para
Simulação [Rio de Janeiro] 1981.

VII, 145p. 29,7cm (COPPE-UFRJ, M.Sc.,
Engenharia de Sistemas e Computação, 1981).

Tese - Univ. Fed. Rio de Janeiro. Programa de
Engenharia de Sistemas e Computação.

1. Simulação 2. Análise 3. Conversacional
4. Teses

I. Univ. Fed. do Rio de Janeiro - COPPE

II. TÍTULO

À minha esposa e filhos.
Aos meus pais.

AGRADECIMENTOS

À USIMINAS - Usinas Siderúrgicas de Minas Gerais S.A. que, através de seu Departamento de Engenharia Industrial, patrocinou este trabalho.

Ao Professor Ricardo Spinelli de Carvalho pela valiosa orientação que me proporcionou na elaboração deste trabalho, ao qual se dedicou intensivamente, durante o período em que trabalhamos juntos.

Ao colega Kleber Alvim Diniz, pelo abnegado e criterioso auxílio na revisão.

À Regina Luiza Silva Garcia e à Marlene Vitor Tuler, pela boa qualidade do material datilografado.

RESUMO

A técnica de simulação é uma ferramenta de Pesquisa Operacional que tem se mostrado bastante útil, versátil e eficiente para a abordagem de problemas razoavelmente complexos. Tem apresentado um avanço considerável, principalmente com o advento dos modernos computadores, proporcionando uma lista extensa de benefícios, tais como, menor esforço de programação, facilidade de localização de erros e geração de resultados, simplificação da expressão de conceitos comuns em simulação, etc.. Observa-se, no entanto, que estes desenvolvimentos estão relacionados muito mais à tradução do modelo e apresentação dos resultados. Existe uma lacuna na parte relativa às facilidades na análise dos resultados, envolvendo as atividades de validação, planejamento estratégico, planejamento tático e experimentação.

O presente trabalho procura mostrar uma metodologia (sistema) para tornar interativa e conversacional algumas das etapas das atividades acima relacionadas, contribuindo desta forma para a economia de recursos (principalmente de tempo) envolvidos num projeto de simulação.

ABSTRACT

The technique of Simulation is an Operational Research tool which appears very useful and efficient for solving complex problems. It has shown remarkable progress mainly due to modern computers, resulting in an extended list of benefits, such as, less programming effort, specific techniques to debug and easy results generation, simplicity to express the usual simulation elements, and so forth. Nevertheless, one may notice that such developments are much more related to the model translation and to the output of results. There is a lack regarding the easiness of analysing simulation results, involving activities such as validation, strategic planning, tactical planning and experimentation.

This work shows a methodology (system) which is able to turn some stages of the activities above mentioned into interactive and conversational operations, thus contributing to the reduction of the resources utilization (mainly, time) involved in a simulation project.

ÍNDICE

	Pág.
CAPÍTULO I - INTRODUÇÃO	1
CAPÍTULO II - ETAPAS DE UM ESTUDO DE SIMULAÇÃO	6
II.1. CONSIDERAÇÕES GERAIS	6
II.2. VALIDAÇÃO	9
II.3. PLANEJAMENTO TÁTICO	10
II.4. PLANEJAMENTO ESTRATÉGICO	15
II.5. EXPERIMENTAÇÃO	16
CAPÍTULO III - LINGUAGENS DE SIMULAÇÃO	18
III.1. CONSIDERAÇÕES GERAIS	18
III.2. GPSS (GENERAL PURPOSE SIMULATION SYSTEM)	19
III.3. GASP IV - GENERAL ACTIVITY SIMULATE PROGRAM - MARK IV	25
III.4. XLSIM2/FOR	30
III.5. SIMSCRIPT II	37
III.6. EXTENDED CONTROL AND SIMULATION LANGUAGE (ECSL)	42
III.7. AS LINGUAGENS E ALGUMAS DE SUAS FUNÇÕES	47

III.8. A ESCOLHA DA LINGUAGEM XLSIM2/FOR	51
CAPÍTULO IV - A SUGESTÃO: O SISTEMA ANALISADOR	53
IV.1. DELINEAMENTO DO SISTEMA	53
IV.2. O SISTEMA ANALISADOR	54
IV.3. ALGUNS EXEMPLOS DE UTILIZAÇÃO DO SISTEMA ANALISADOR	62
IV.4. CONSIDERAÇÕES FINAIS	120
CAPÍTULO V - CONCLUSÕES FINAIS E SUGESTÕES	121
BIBLIOGRAFIA	124
APÊNDICE I	128

CAPÍTULO I

INTRODUÇÃO

O rápido desenvolvimento verificado nas últimas décadas, em termos de equipamentos e facilidades relacionadas ao processamento de dados, contribuiu de forma decisiva para que se ampliassem as potencialidades de diversos métodos quantitativos de apoio à tomada de decisões, entre os quais a simulação. Estimulados por perspectivas cada vez mais amplas, vários institutos de pesquisas, centros universitários e empresas empenharam-se em desenvolver aplicações e introduzir refinamentos metodológicos que lograram transformar as técnicas de simulação em uma ferramenta útil, versátil e eficiente para a abordagem de problemas razoavelmente complexos, conforme se pode atestar pelos resultados de recente pesquisa realizada entre as 1.000 maiores empresas americanas quanto à utilização de métodos quantitativos em planejamento estratégico¹:

METODOLOGIA	% DAS APLICAÇÕES
Simulação	29
Programação Linear	21
Análise de Redes (Pert)	14
Teoria de Estoques	12
Programação Não Linear	8
Programação Dinâmica	4
Programação Inteira	3
Teoria das Filas	3
Outros	6

No início, as publicações na área de simulação se concentraram no desenvolvimento de geradores de números aleatórios e variáveis aleatórias. Uma excelente pesquisa cobrindo o período anterior a 1962 é apresentada no trabalho de Hull e Dobell,² que contém cerca de 150 citações bibliográficas. Outra compreensiva matéria sobre o assunto está contida no trabalho de Jansson,³ que contém cerca de 300 referências.⁴

Os primeiros projetos realizados usando simulação foram generalizações de modelos de problemas específicos de considerável importância militar e econômica. Exemplos típicos são "Job Shop Simulation", "Inventory Management Simulation"⁵ e numerosas simulações de sistemas operacionais e militares.

Os conhecimentos obtidos nestes trabalhos contribuíram grandemente para o desenvolvimento da simulação, com o aparecimento das linguagens gerais e dos programas de simulação, cujas estruturas se desenvolveram segundo as seguintes linhas de pesquisa⁶:

a) Abordagem por Eventos (Linha Americana): - um evento é, à luz da abordagem por eventos, uma descrição dos passos que ocorrem em consequência de uma alteração no estado de uma entidade. Por exemplo, quando uma máquina (entidade permanente) pára de funcionar (mudança de estado) a sua possível reutilização é imediatamente investigada. A simulação consiste de um número desses eventos, permitindo, assim, que seja montada uma lista dos eventos programados para ocorrer em ordem cronológica simulada. Daí, também, a designação usual do próximo evento para esta abordagem.

b) Abordagem por Atividades (Linha Inglesa): - uma atividade consiste de duas partes: um teste inicial e uma descrição dos passos que ocorrem em consequência, no caso de todos os requerimentos terem sido satisfeitos, o que indica que a atividade pode ser executada. O mecanismo do tempo, em contrapartida ao controle do tempo na técnica de eventos, função do momento de ocorrência do próximo evento, é aqui determinado

pela comparação entre os tempos em que cada entidade ainda deverá permanecer em seu presente estado. O avanço do relógio é dado pelo menor destes tempos.

c) Abordagem Mista (Eventos-Atividades): modelagem em três fases (clássica de Tocher⁷) - na primeira fase é feita a seleção do próximo evento programado, movendo-se o relógio até o instante de ocorrência desse evento; na segunda, completam-se as ações relativas àquele evento selecionado e de outros eventos porventura coincidentes naquele instante sendo simulado; na terceira fase checam-se todas as atividades correlacionadas com os eventos múltiplos e iniciam-se as mesmas, quando possível; quando todos os testes são concluídos, retorna-se à primeira fase.

Pesquisas recentes têm apresentado à comunidade da simulação, a técnica de simulação celular, que é uma forma de modelagem que agrupa as atividades do sistema em células, de forma a aumentar a eficiência da experimentação com o modelo.^{8,9,10} Este tipo de abordagem permite, numa dada corrida, que determinada(s) célula(s) do sistema seja(m) ou não simulada(s), com base no valor de parâmetros obtidos em corridas anteriores. Caso não seja(m) simulada(s) aquela(s) célula(s), suas saídas, anteriormente registradas, são suprimidas do resto do modelo, como se fosse(m) ocorrência(s) exógena(s).

Por outro lado, quando os computadores eram lentos, os pesquisadores encontraram-se numa posição onde era muito caro decrescer a variabilidade dos resultados da simulação aumentando-se o tamanho da corrida. Como resultado, um crescente interesse no desenvolvimento dos procedimentos de amostragem e estimação levaram a um aumento na precisão dos estimadores para um tamanho de corrida fixo ou reciprocamente um decréscimo no tamanho da corrida requerido para a obtenção de um grau fixo de precisão. Estes procedimentos ficaram conhecidos como Técnicas de Redução de Variância.⁴

Os esforços recentes têm chegado aos quase-geradores (linguagem simbólica, bastante simplificada, embora não seja transferida para a máquina, como nos geradores; exemplos: GPSS^{11,12}, Q-GERT¹³) e aos geradores (interface geral entre os modelos de simulação e linguagens ou programas de simulação; exemplos: DRAFT¹⁴, CAPS¹⁵, GERASIM²¹). Estes esforços estão concentrados naquilo que tem mais exigido reflexos da técnica de simulação: - um bom modelo de simulação demanda razoáveis recursos em termos de tempo, dinheiro e conhecimento da técnica e de suas dificuldades, desde a formulação do problema até a interpretação dos resultados.

No entanto, apesar de todos estes avanços proporcionarem uma lista extensa de benefícios, tais como: exigência de menor esforço de programação, técnicas específicas de localização de erros, simplificação da expressão e articulação de conceitos comuns em simulação, geração de resultados com facilidade, apresentação padronizada destes resultados, eficiente alocação de memória, etc, observa-se que estes desenvolvimentos estão relacionados muito mais à tradução do modelo e geração de resultados. Existe uma lacuna na parte relativa às facilidades na análise dos resultados, envolvendo as atividades de validação, planejamento estratégico, planejamento tático e experimentação.

O presente trabalho procura mostrar uma metodologia para tornar interativa as atividades acima relacionadas, proporcionando deste modo uma ligação com os geradores e permitindo, num futuro próximo, o desenvolvimento de um estudo de simulação, desde a definição do sistema (formulação do problema) até a interpretação dos resultados, todo ele de forma conversacional, contribuindo desta forma para a economia de recursos envolvidos no projeto.

Os capítulos seguintes procurarão mostrar o desenvolvimento que nos levou a sugerir a metodologia apresentada. O capítulo II versará sobre as etapas de um estudo de simulação de um modo geral, salientando-se aquelas cobertas pela metodologia proposta; o capítulo III mostrará diversas linguagens e

suas facilidades dentro do enfoque estabelecido no capítulo II; o capítulo IV apresentará a metodologia propriamente dita, enquanto que no capítulo V apresentaremos as conclusões e algumas sugestões para pesquisas futuras; finalmente, no apêndice I, será apresentada a listagem do programa.

CAPÍTULO II

ETAPAS DE UM ESTUDO DE SIMULAÇÃO

II.1. CONSIDERAÇÕES GERAIS

Uma vez decidido que a solução de um problema será obtida através de um estudo de simulação, o analista deve seguir uma série de tarefas concatenadas, de cuja realização depende a obtenção de resultados significativos. A maioria dos autores, entre eles Shannon,¹ sugere as seguintes etapas:

1) - DEFINIÇÃO DO SISTEMA: determinação dos objetivos, condições, restrições e medidas de eficiência que caracterizam o sistema a ser estudado.

2) - FORMULAÇÃO DO MODELO: redução ou abstração do sistema real a um diagrama lógico de blocos e/ou fluxogramas.

3) - PREPARAÇÃO DOS DADOS: identificação dos dados necessários ao modelo e redução destes a uma forma apropriada para sua utilização.

4) - TRADUÇÃO DO MODELO: descrição do modelo em uma linguagem aceitável pelo computador a ser utilizado.

5) - VALIDAÇÃO: determinação da validade do modelo, de modo que inferências a partir do modelo, com relação ao sistema real, serão consideradas corretas, dentro de um aceitável nível de confiança.

6) - PLANEJAMENTO ESTRATÉGICO: projeto de um experimento que irá conduzir à(s) informação(ões) desejada(s).

7) - PLANEJAMENTO TÁTICO: determinação de como cada corrida do projeto de experimento deve ser executada.

8) - EXPERIMENTAÇÃO: execução da simulação para gerar os resultados desejados e realizar análises de sensibilidade.

9) - INTERPRETAÇÃO: fazer inferências a partir dos resultados gerados pela simulação.

10) - IMPLEMENTAÇÃO: colocar o modelo e/ou os resultados em uso.

11) - DOCUMENTAÇÃO: registrar as atividades do projeto e os resultados, bem como o modelo e seu uso.

Estas etapas e suas relações são mostradas na figura 1 da página seguinte.

Embora não seja esta a única forma de desmembrar um projeto de simulação e possa não corresponder, cronologicamente, ao que se observa na prática, serve de modo claro para mostrar a necessidade de uma metodologia que permita fazer uso integral dessa poderosa ferramenta colocada à disposição dos administradores de complexas organizações humanas.

Como já foi dito, nossa proposta pretende tornar as etapas de 5 a 8 parte integrante de um projeto maior, que é tornar a simulação um procedimento interativo, facilitando, desta forma, ainda mais o seu desenvolvimento, diminuindo o tempo e esforço despendido na sua realização, que é uma das suas facetas mais criticadas. Deste modo, analisaremos a seguir, de uma forma sucinta, estas etapas, procurando orientá-las para o nosso propósito, já que seria difícil generalizá-las. Também não é nossa intenção discutí-las, já que existem diversos trabalhos neste campo. Procuraremos apenas repassar o que já foi escrito a respeito do assunto, realçando o que for importante para o nosso propósito.

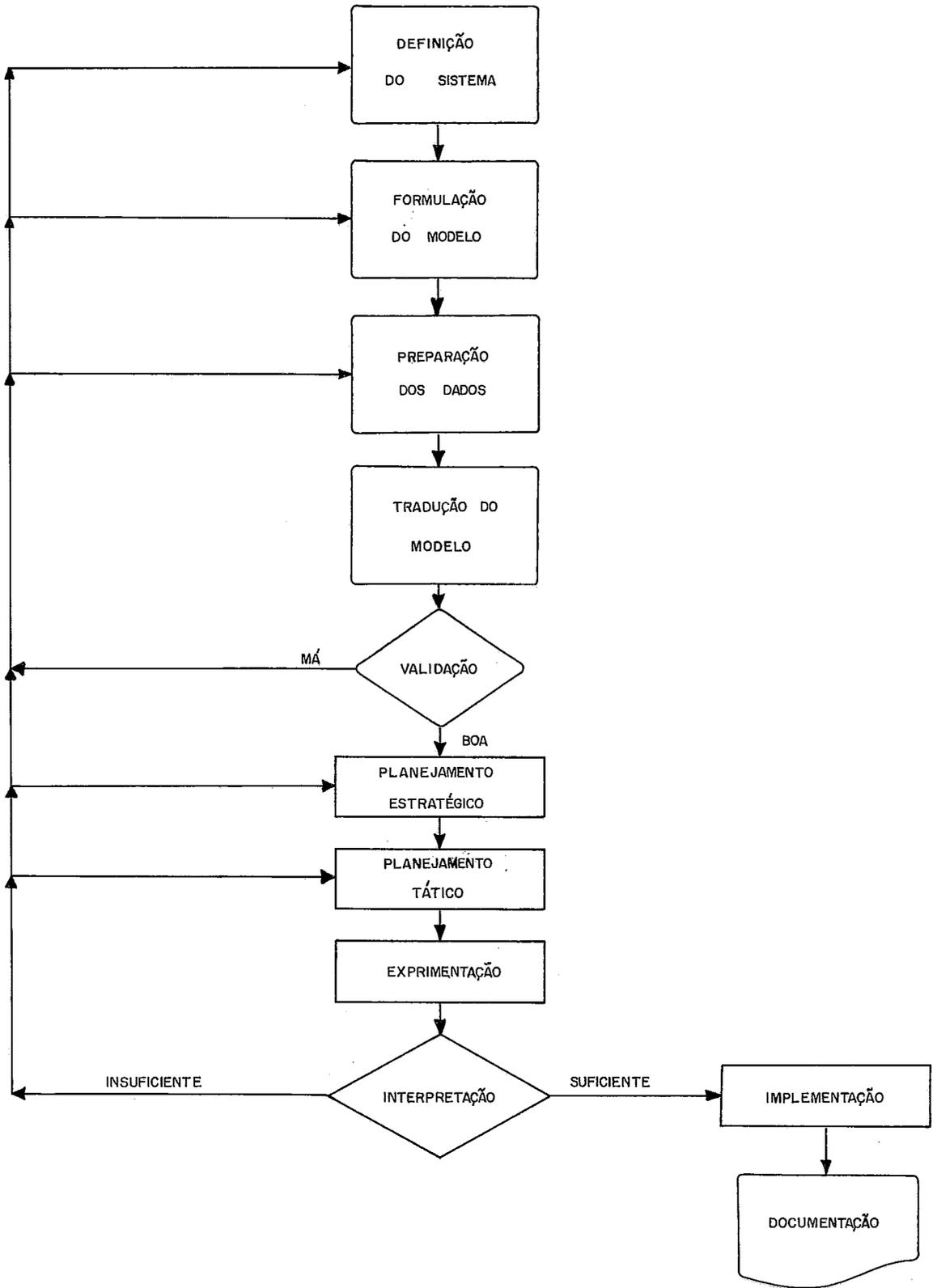


Figura 1 - Etapas de um Estudo de Simulação

II.2. VALIDAÇÃO

Um modelo é criado para um determinado propósito e a sua adequação e validação só podem ser desenvolvidas em termos deste propósito. Elaborar um modelo significa desenvolver um aceitável nível de confiança, de modo que inferências a serem extraídas dos resultados do modelo sejam corretas e aplicáveis ao sistema real. Validação é um processo contínuo, que começa desde o início do estudo. A confiança ou credibilidade do modelo desenvolvido cresce ao longo do estudo. A melhor validação possível será alcançada se, entre outras coisas⁴:

- 1) - usarmos sempre lógica e bom senso;
- 2) - utilizarmos o conhecimento e inspiração daqueles que são familiarizados com o sistema real;
- 3) - testarmos empiricamente (através das técnicas estatísticas apropriadas) todas as suposições, hipóteses, etc, que possam ser testadas;
- 4) - prestarmos atenção aos mínimos detalhes, checando cada passo do processo de modelagem;
- 5) - estivermos certos que o modelo se comporta da maneira que desejamos (testes na depuração);
- 6) - a transformação "input-output" do modelo e a do sistema real são comparáveis;
- 7) - executarmos testes de campo, quando possível;
- 8) - realizarmos análises de sensibilidade das variáveis de entrada, parâmetros, etc;
- 9) - checarmos detalhadamente as previsões do modelo e os atuais resultados obtidos com o sistema real.

Dentro deste enfoque e, não abandonando o ponto de vista de que a validação sempre depende do tipo de problema,

do propósito do modelo, dos custos envolvidos e de fatores subjetivos, pode-se procurar estabelecer algumas linhas gerais para um esquema de validação, proposta inicialmente por Naylor e Finger¹⁶, que são:

- a) crítica dos subsistemas do modelo e das hipóteses subjacentes;
- b) teste empírico das hipóteses explicitamente consideradas;
- c) comparação de "input-output" entre o modelo e a realidade.

Assim sendo, através: - de uma depuração lógica, procuramos assegurar que o modelo realiza aquilo que se pretende representar; da impressão de estatísticas dos resultados, criticamos racionalmente e sistematicamente o modelo, procurando a exclusão das variáveis irrelevantes, a representação adequada de variáveis e a formulação correta da relação entre as variáveis; e, quando possível, comparação "input-output", através de testes estatísticos apropriados (tais como, Kolmogorov-Smirnov, Qui-quadrado, etc.)¹⁷.

II.3. PLANEJAMENTO TÁTICO

Desde que a variabilidade é uma característica de todos os modelos de simulação estocástica, reaplicações são necessárias para a obtenção de resultados precisos e exatos. O analista deve conduzir seu experimento de tal modo a não somente obter os resultados, mas também estimar a precisão dos resultados e o nível de confiança atribuído, para as conclusões extraídas destes resultados. Entre as questões que precisam ser aplicadas, e que são comumente chamadas de problemas táticos¹⁸, temos: condições iniciais, convergência, auto-correlação, tamanho da simulação e técnicas de redução de variância.

1) - Condições Iniciais e Convergência: - A primeira dificuldade a ser considerada é a de como minimizar o efeito de desequilíbrios gerados no início de operação de um modelo. A razão de tais desequilíbrios reside no próprio conhecimento incompleto do sistema em estudo, que faz com que, em todo ou em parte, as condições iniciais impostas possam corresponder a um estado atípico do sistema. Conway¹⁸ observa que "o ideal seria relacionar condições iniciais por um processo aleatório cujas probabilidades correspondessem às da distribuição de probabilidades em regime estacionário". Entretanto, caso isto fosse possível, a simulação torna-se-ia possivelmente desnecessária. Há pelo menos três métodos que podem ser usados para reduzir o efeito do período transiente inicial de modo a minimizar o efeito tendencioso que ele possa ter sobre os resultados:

a) usar corridas longas de modo que os resultados do período transiente sejam insignificantes, com relação aos resultados da condição de equilíbrio: - para computadores rápidos, operando modelos simples, esta alternativa pode ser preferida, não sendo, porém, razoável sob outras condições;

b) eliminar, para efeito de cálculo das estatísticas desejadas, os resultados referentes aos períodos iniciais, que produzem mais distorção do que informação: - a dificuldade ao se adotar esta alternativa é determinar que extensão de período deve ser eliminada. Se for muito pequena, não atenderá o objetivo pretendido; se for demasiadamente grande, elevar-se-ão os custos de resultados desperdiçados, ampliando-se a variância dos resultados finais, uma vez que esta é inversamente proporcional à raiz quadrada do número de observações, e provavelmente cairemos na situação a. Tal equilíbrio entre a escolha de um período pequeno e um período grande não é, entretanto, de fácil determinação teórica, o que torna mais adequado escolher-se uma regra prática aceitável para o truncamento das observações, tendo sempre o cuidado de verificar o

eventual prejuízo causado em termos de aumento na variância dos estimadores. Uma destas regras práticas pode ser a seguinte¹⁹:

- observar a evolução das médias das estatísticas mantidas; para isto deve-se executar, a intervalos convenientes, uma rotina que compreenda o registro da média de cada estatística e o esvaziamento das mesmas (zerando-as), pois o uso de estatísticas cumulativas com esse fim, levaria a uma falsa análise, caracterizada pelo tardio reconhecimento da situação de regime estacionário; os valores destas médias podem ser plotados, em relação ao tempo, constituindo um recurso visual de grande valia para a determinação do período transiente. Certamente, um critério matemático mais preciso pode ser desenvolvido com base nestes mesmos dados, de forma que, a partir da relativa estabilização das médias, possa se considerar atingido o regime estacionário.

Qualquer que seja a regra prática escolhida para determinar o período transiente, é necessário que se disponha de um mecanismo capaz de levar a efeito o truncamento, quando da execução das corridas desejadas; neste truncamento devem ser eliminados todos os registros até então feitos, conservando-se apenas as informações sobre o estado atual do sistema, que passarã a ser, efetivamente, o estado inicial a partir do qual o modelo será operado.

c) a terceira alternativa, não exclusiva em relação às primeiras, é a simples escolha adequada das condições iniciais: condições iniciais típicas podem encurtar sensivelmente o período transiente, às custas de um trabalho adicional para determiná-las e impô-las; entretanto, esta alternativa oferece alguns inconvenientes: primeiro, o trabalho adicional e segundo, o perigo de se introduzirem condições iniciais incompatíveis, incompletas ou excessivamente severas que venham a prejudicar a operação do modelo.

Como vemos, cada uma destas sugestões apresentam os seus próprios problemas em termos de implementação e ainda

permanece um fértil assunto para pesquisas futuras.

2) - Auto-Correlação: - nem sempre é possível aceitar a hipótese de que as sucessivas observações de uma variável em um modelo de simulação são estatisticamente independentes. Tal suposição, que é extremamente conveniente do ponto de vista de análise de resultados, só será válida em alguns modelos simples. Na maior parte dos problemas reais, entretanto, o resultado de uma observação em dado instante apresentará forte dependência em relação ao estado do sistema nos instantes das observações anteriores. Neste caso se diz existir auto-correlação entre os resultados, significando que a informação acrescentada para cada nova observação será menor do que se estas fossem independentes. Embora a presença de auto-correlação não afete os estimadores de médias dos parâmetros de interesses, terá um efeito sensível sobre os estimadores de variância. Para contornar essa dificuldade pode-se optar por um dentre os procedimentos distintos:

a) estimação explícita da função de auto-correlação e a inclusão de seus efeitos na estimação dos parâmetros: - este procedimento exige a realização de uma corrida piloto, para que se possa estimar as correlações amostrais $\rho(s)$ entre observações defasadas de s unidades de tempo.

b) a outra alternativa comum para contornar o problema de auto-correlação é o agrupamento dos dados em blocos: - este método consiste em não levar em consideração os valores individuais observados, mas sim o valor médio obtido de cada bloco de observações; mesmo que haja uma forte auto-correlação entre os valores da variável de interesse, ao se agruparem diversas observações da mesma haverá uma diluição deste efeito permitindo considerar o valor médio obtido em cada bloco como se fosse uma única observação, praticamente independente das demais.

3) - Tamanho da Corrida: - a escolha de um apropriado tamanho da corrida de simulação (tamanho da amostra) para obter o desejado nível de precisão e também minimizar o custo de operação do modelo é uma tarefa difícil e extremamente importante. Considerando que a informação fornecida pelo experimento de simulação será usada como base para decisões relativas à operação do sistema real, esta informação será tão acurada e precisa quanto possível, ou ao menos, com um insignificante grau de imprecisão. É portanto essencial que uma análise estatística seja conduzida para determinar o tamanho da simulação requerido. Este tamanho de corrida pode ser determinado da seguinte maneira:

a) a priori e independentemente da operação do modelo;

b) durante a operação do modelo e baseado nos resultados gerados pelo modelo.

A capacidade de adotar a alternativa (a) dependerá da existência de informação prévia a respeito do comportamento do modelo. A alternativa seguinte é determinar os intervalos de confiança para os resultados do modelo, de acordo como eles são gerados pelo modelo, durante a corrida de simulação e então terminar a execução quando um intervalo de confiança pré-determinado tenha sido obtido.

4) - Técnicas de Redução de Variância: - estas técnicas foram muitas vezes referidas como técnicas de Monte Carlo⁴, mas desde que o uso deste termo causou grande confusão, o termo técnicas de redução de variância tem-se tornado amplamente aceito. Estas técnicas visam aumentar a precisão dos estimadores sem que se precise aumentar o tamanho da corrida. Dentre as mais discutidas, podemos citar: amostragem estratificada, amostragem de importância, variável de controle, medidas virtuais, variáveis antitéticas e amostras correlacionadas. Estas técnicas foram propostas e desenvolvidas no período com-

preendido entre os primórdios de 1940 e primórdios de 1960. É natural e não por acidente que o interesse nestas técnicas era alto, quando os computadores eram lentos e declinou quando a velocidade de processamento dos computadores aumentou. Contudo, como as simulações têm aumentado em complexidade, isto vem contrabalançando o acréscimo de velocidade e existe um renovado interesse no uso destas técnicas. Os estudos feitos mostram que, quando estas técnicas são aplicadas a problemas simples, os resultados apresentados pelos modelos são bastante precisos. Com modelos mais complexos, de problemas reais, os ganhos obtidos, embora significantes, não são tão impressionantes.

II.4. PLANEJAMENTO ESTRATÉGICO

O projeto de experimentos é um tópico cuja relevância para a simulação tem sido longamente reconhecida, mas raramente aplicada na prática. O projeto de um experimento de simulação por computador é essencialmente um plano para obter-se uma quantidade de informação que pode ser adquirida de várias formas, dependendo do modo pelo qual os resultados são obtidos. Os propósitos de um projeto de experimento são: - econômicos, em termos de reduzir o número de experimentos necessários; - didáticos, quando providencia uma estrutura para o processo de aprendizagem do investigador. A corrida de um experimento em simulação é o processo de correr o modelo de modo a observar e analisar as informações resultantes para a obtenção da resposta desejada. O projeto de experimento também identifica uma particular abordagem para a obtenção desta informação, de modo a permitir inferências válidas sobre os resultados alcançados. Dependendo do propósito específico do experimentador, existem vários tipos de análises que podem ser requeridas. Entre as mais comuns, estão:

- comparação de médias e variâncias de alternativas;

- determinação da importância ou efeito de diferentes variáveis e suas limitações;
- pesquisa de valores ótimos de um conjunto de variáveis.

Os projetos de experimentos para o primeiro tipo de análise são geralmente chamados experimentos a um único fator e estão diretamente relacionados com assuntos tais como tamanho de corrida, condições iniciais e auto-correlação. O segundo tipo de abordagem utiliza análise de variância e de regressão para interpretação dos resultados. O terceiro tipo de análise requer usualmente técnicas de pesquisa.

II.5. EXPERIMENTAÇÃO

Um modelo de simulação é construído a fim de que se possam conduzir experimentos controlados que seriam impossíveis, ou impraticáveis, ou ainda economicamente proibitivos de serem efetuados em um sistema real complexo. A experimentação conduzida através de um modelo de simulação se insere na categoria mais geral da investigação experimental. Deve-se notar ainda que modelos de simulação são em geral utilizados em situações nas quais não existem relações funcionais simples, caso em que técnicas específicas de otimização, tais como programação linear, seriam mais adequadas.

Em alguns casos o número de alternativas a serem investigadas é pequeno; o método de busca de uma solução ótima se reduz então à verificação e comparação de todas as possibilidades. Em outros casos, contudo, o número de variáveis de decisão ou controles envolvidos e os diversos valores possíveis para cada um podem tornar inviável a enumeração completa das alternativas. Neste caso, a busca de uma otimização, ou mesmo a caracterização de uma superfície de resposta passam a constituir tarefas não triviais. O procedimento a ser adota

do pode consistir em realizar uma série de corridas com o modelo de simulação, efetuando-se mudanças pré-determinadas nas variáveis de decisão, entre uma corrida e outra.

A alternativa que se apresenta é, em vez de pré-fixar uma sequência de busca, adotar um procedimento iterativo, em que os valores das variáveis de decisão a serem usados em cada nova corrida de simulação são decididos pelo experimentador, com base nos resultados das corridas anteriores. O pesquisador pode assim orientar o experimento no sentido que lhe parecer mais adequado. Entretanto, se o número de iterações máquina-homem-máquina, for razoavelmente grande, haverá uma considerável perda de tempo em procedimentos de entrada e saída, mesmo que sejam utilizados equipamentos periféricos modernos. Este inconveniente pode ser contornado, desde que o pesquisador seja capaz de formalizar o processo mental através do qual, com base em um conjunto de resultados já obtidos, decide se o experimento deve prosseguir ou não e, em caso afirmativo, para quais valores devem ser modificados os parâmetros do modelo.

CAPÍTULO III

LINGUAGENS DE SIMULAÇÃO

III.1. CONSIDERAÇÕES GERAIS

As "linguagens" escolhidas para a abordagem de seu comportamento dentro do enfoque mostrado no capítulo II são as que, dentro da simulação discreta, abrangem os diversos conceitos gerais de simulação. As características a serem apresentadas devem ser entendidas como sendo as dos diversos segmentos, não refletindo vantagem de uma dada linguagem sobre qualquer outra. Por outro lado, também não é nossa intenção ser completo na citação das linguagens e suas funções específicas, o que seria impossível, já que existem, ^{4,22} só nos Estados Unidos da América, mais de 170 linguagens ²⁸, e novas estão sendo desenvolvidas a cada dia ²⁸.

O XLSIM2/FOR ^{23,36} e o GASP IV ²⁴ são considerados programas de simulação, embora designados genericamente como linguagens. Não dispõem de sintaxe e compilador próprios e são constituídos de um conjunto de sub-rotinas e funções em FORTRAN. Possuem poucos mas poderosos comandos específicos de simulação e geralmente são de fácil aprendizado, dado serem apoiados em linguagem (FORTRAN) bastante conhecida, permitindo a sua plena utilização para a modelagem de qualquer sistema.

O XLSIM2/FOR ⁷ baseia-se na técnica de três fases, devido a Tocher ^{8,9,10}, permitindo utilizar o conceito de simulação celular ^{8,9,10}, facilitando a experimentação e não apenas a modelagem.

O GASP IV é baseado na abordagem por eventos, quando voltado para simulação discreta. Permite também simulação contínua e combinada (contínua-discreta), fora do escopo deste trabalho. Possui versão também em PL/I.

O SIMSCRIPT II^{5,25,29,30} e o ECSL¹⁵ são considerados "línguas de simulação", pois possuem sintaxe e compilador próprios. Sendo línguas específicas de simulação, têm mais recursos que os programas de simulação acima mencionados; a programação dos modelos é ainda mais fácil, logicamente.

O ECSL é uma linguagem de varredura de atividades (activity-scan), enquanto que a estrutura da linguagem SIMSCRIPT II é a do próximo evento, permitindo também a programação por varredura de atividades ou ambas.

O GPSS^{5,11,12,29,30} é considerado um quase-simulador: - esta designação deve-se a não ser ele tão restrito quanto os simuladores (que são pacotes escritos numa determinada linguagem e que se destinam a uma finalidade específica⁶). Sua programação, em linhas gerais, é bastante simplificada, embora não seja transferida para a máquina como nos geradores. Trata-se de linguagem caracterizada por um fluxograma de blocos lógicos, isto é, um diagrama que representa o fluxo de transações através de certos blocos pré-definidos, cada qual representando uma instrução da linguagem.

III.2. GPSS (GENERAL PURPOSE SIMULATION SYSTEM)

III.2.1. INTRODUÇÃO

A linguagem GPSS é destinada à resolução de problemas de simulação de modelos discretos. É descrita sob a forma de diagrama de blocos, que representam as atividades, e de linhas que unem os blocos indicando a sequência na qual estas atividades devem ser executadas. Quando mais de uma linha deixa o bloco, a condição de escolha é decidida pelo bloco. Cada bloco tem um significado preciso e é representado por um símbolo particular. O GPSS possui um compilador próprio.

III.2.2. CONSIDERAÇÕES GERAIS

O GPSS é uma linguagem de diagrama de blocos, cada um representando uma ação particular. O usuário deve descrever as "ATIVIDADES" (ou fatos que acontecem) no sistema por meio destes tipos básicos de blocos.

O GPSS é uma linguagem "ORIENTADA PARA PARTÍCULAS", isto é, sua atenção se concentra nas "ENTIDADES" (um objeto de interesse no sistema, tal como um cliente de um supermercado) e vai seguindo as mudanças que ocorrem quando as entidades se movem de atividade para atividade. Chamamos de "ATIVIDADE" um processo que modifica o estado do sistema (tal como a chegada, o atendimento ou a saída de um cliente).

As entidades são classificadas como temporárias e permanentes. As entidades temporárias são criadas durante a simulação, se movem pelo sistema e são, finalmente, removidas do sistema (por exemplo, um cliente em um supermercado). O GPSS lhes dá um nome especial: "TRANSAÇÕES". As entidades permanentes são elementos do sistema e permanecem as mesmas durante a simulação (por exemplo, o número de carrinhos de um supermercado). São chamados de "EQUIPAMENTOS". De modo a melhor fixar estes últimos conceitos, apresentamos alguns exemplos no Quadro 1.

SISTEMA	TRANSAÇÕES	EQUIPAMENTOS	ATIVIDADES
Cab. Telefônica	Pessoas	Telefone	Telefonema
Supermercado	Pessoas	Carrinhos/Caixa	Compra
Tráfego	Carros	Semáforos/ Quarteirões	Movimento

Quadro 1 - Aplicações dos Conceitos de Simulação em GPSS

Assim sendo, para aplicar o GPSS em um sistema qualquer é necessário que todos os elementos do sistema sejam identificados com elementos do GPSS. Devemos definir o que são transações no sistema. Devemos identificar os equipamentos e verificar sua capacidade. Devemos descrever as atividades em termos dos blocos existentes.

Uma vez que tenhamos adaptado o nosso modelo à "Visão-do-mundo" do GPSS, estamos aptos a usá-lo.

Os elementos chaves de um modelo em GPSS são "blocos", "transações" e "equipamentos" (facilities/storages). Os modelos de simulação em GPSS são desenvolvidos em termos de diagramas de blocos, os quais são dispositivos gráficos, que retratam o fluxo lógico e físico das transações através do sistema.

Transações, que são os elementos temporários do GPSS, se movem, de um bloco a outro, no modelo. Existem 8 (oito) parâmetros associados com cada transação e 8 (oito) possíveis níveis de prioridades. Quando uma transação entra num bloco, a sub-rotina associada com este particular tipo de bloco é executada, modificando um ou mais descritores de estado.

Para representação de equipamentos, o GPSS possui três tipos de elementos: "facilities", "storages" e "logic switches", que podem ser usados para representar equipamentos físicos ou conceitos lógicos.

Transações, que chegam num equipamento, são atendidas na ordem de chegada de acordo com a classe de prioridade. Existe um total de 10 (dez) tipos de elementos no GPSS, cada um designado por um ou mais tipos de blocos:

Elementos básicos: blocos; transações.

Elementos de equipamentos: "facilities"; "storages"; "logic switches".

Elementos estatísticos: filas; tabelas de distribuição.

Elementos de referência: "savexes".

Elementos computacionais: variáveis aritméticas; funções.

Dos numerosos atributos associados com estes tipos de elementos, há um subconjunto de 15, chamados "variáveis padrões do sistema" (Standard System Variables), cujos valores podem ser obtidos num modelo de simulação pelo nome e índice numérico; por exemplo, Q13 é o conteúdo corrente da fila 13 (um número inteiro indicando o número de transações na fila); S7 é o conteúdo corrente da "storage" 7 (também um número inteiro); FN35 é o valor corrente da função 35.

Praticamente toda troca de estado nos modelos em GPSS ocorre como resultado da chegada de uma transação em diferentes tipos de blocos, executando a sub-rotina associada com o tipo do bloco. Uma variedade de procedimentos chamados "selection modes" é usada para estabelecer a escolha do próximo bloco para o qual a transação irá se deslocar.

Novas transações são criadas e entram no modelo em GPSS através dos blocos "ORIGINATE" e "GENERATE". As transações são destruídas nos blocos "TERMINATE" e "ASSEMBLE".

Quando uma transação entra num bloco "SPLIT", uma transação idêntica é criada. Tais transações são identificadas como membros de "transactions assembly sets" e podem ser manipuladas com os blocos "ASSEMBLE" e "MATCH". Transações num dado modelo podem ser registradas numa especificada fita magnética através dos blocos "WRITE". Estas transações podem ser subsequentemente lidas desta fita, dentro de qualquer bloco num outro modelo GPSS.

"Facilities" são usadas nos modelos em GPSS para representar equipamentos com capacidade unitária. "Storages" são usadas para representar equipamentos com capacidades maior que um. Consequentemente, muitas transações podem mover-se nos blocos "ENTER", referindo-se a uma particular "storage" e desta forma aumentando a quantidade de itens naquela "storage".

As transações que entram nos blocos "LEAVE" reduzem a quantidade de itens na "storage". Quando a quantidade de itens de uma "storage" é igual a uma capacidade pré-definida, nenhuma transação, posteriormente, pode mover-se dentro de blocos "ENTER" associados com aquela "storage".

Os blocos "QUEUE" podem ser inseridos em um modelo em GPSS depois dos blocos nos quais as transações podem se recusar a entrar, como os blocos "SEIZE" e "ENTER". Cada bloco "QUEUE" se refere a uma fila particular, cuja função principal é reunir as estatísticas a respeito da quantidade de itens e tempo de ocupação.

As variáveis aritméticas, que são definidas pelos cartões "VARIABLE", expressam combinações aritméticas, parecidas com o FORTRAN das variáveis padrões do sistema. Todas as operações são em valores inteiros, e um operador modular é fornecido juntamente com os operadores aritméticos padrões.

Funções, que são definidas pelos cartões "FUNCTION", expressam um relacionamento funcional entre uma variável independente, a qual pode ser qualquer uma das variáveis padrões do sistema, e um valor da função da variável dependente, FN_j , a qual é uma das variáveis padrões do sistema. Cada função é definida por n pares de pontos.

As transações podem demorar um tempo específico em qualquer bloco. Estas transações desaparecem dentro da cadeia de eventos futuros na ordem crescente do tempo inicial destes blocos. A maioria das transações que permaneceram, são referidas na "cadeia de eventos correntes", na qual elas são ordenadas de acordo com o tempo que foram programadas, dentro de cada classe de prioridade. O sequenciamento dinâmico de eventos é providenciado por um mecanismo de varredura (Scan), o qual, a cada tempo relógio (clock time) examina a cadeia de eventos correntes, tentando fazer as transações se moverem para um próximo bloco possível. Quando o mecanismo do GPSS move uma transação para um próximo bloco, ele tenta manter a transação movimentando-se através de tantos blocos quanto seja possível, até que uma das três coisas seguintes aconteça:

1) - a transação encontra um tempo de demora positivo no bloco e é eliminada pela cadeia de eventos futuros;

2) - a transação é finalmente bloqueada na entrada de um bloco, permanecendo aí;

3) - a transação é destruída no bloco "TERMINATE" ou "ASSEMBLE".

O GPSS concentra-se quase que exclusivamente em processar uma simples transação que se esteja movendo no momento. As outras transações são afetadas somente indiretamente. Eventualmente, o GPSS faz uma verificação completa na ordem de eventos correntes, mas sem intenção alguma de mover qualquer transação para o próximo bloco. O tempo do relógio é então modificado para o tempo de saída do bloco da primeira transação. A primeira transação e todas as outras transações com o mesmo tempo de saída do bloco são transferidas para a cadeia de eventos correntes. O GPSS começa outra vez a tentar mover as transações da cadeia de eventos correntes para os próximos blocos.

Um conjunto padrão de dados estatísticos é escrito num registro de saída padrão no fim de corrida da simulação. Os cartões de controle "START" especificam a duração de cada corrida, assim como também especificam o número de transações que devem ser destruídas nos blocos "TERMINATE". Durante cada corrida, o GPSS acumula um conjunto de estatísticas para cada variável usada no modelo. Estas estatísticas incluem o tempo acumulativo de ocupação de uma entidade qualquer e a quantidade de transações que a ocuparam. Entre as estatísticas estão: a média e o número máximo; o tempo médio gasto pelas transações dentro da entidade; a utilização média de uma disponibilidade ou capacidade de armazenagem do sistema.

Quando a transação entra no bloco "TABULATE", que se refere a uma tabela de uma distribuição particular, o valor do argumento da tabela, que pode ser uma das variáveis padrões do sistema, é tabulado. Cada tabela nos dá o valor médio e o desvio padrão dos valores tabulados do argumento da tabela. Além disso, as frequências absoluta e relativa dos valores das variáveis são acumuladas e impressas.

O GPSS também fornece como saída padrão uma contagem do número total de transações que entraram em cada bloco durante a corrida; e também o número de transações correntes em dado bloco no fim da corrida.

III.2.3. FORMULAÇÃO DE MODELOS EM GPSS

O processo de traduzir um problema em um programa GPSS inicia-se com a decisão de se aplicar GPSS à solução do problema considerado.

Após o estabelecimento da configuração básica do sistema e de suas regras de operação, o próximo passo é formular um modelo em GPSS. Para tanto, um modelo preliminar deve ser formulado em termos das entidades GPSS, em forma de fluxograma ou diagrama de blocos (blocos GPSS). Após formulado este diagrama, passa-se a escrever o programa em GPSS. Consiste de uma sequência de instruções (comandos) que mais ou menos duplicam os elementos do diagrama de blocos. Estes comandos são, geralmente, de quatro tipos:

- 1) Comandos de especificação (Blocks types);
- 2) Comandos de definição;
- 3) Comandos de controle do programa;
- 4) Comandos de comentário.

GPSS é uma linguagem de alto nível: cada instrução faz o computador executar um grande número de tarefas ou, em outras palavras, cada comando GPSS usualmente executa uma ou mais sub-rotinas.

III.3. GASP IV - GENERAL ACTIVITY SIMULATE PROGRAM - MARK IV

III.3.1. INTRODUÇÃO

GASP IV fornece uma "visão do mundo" que simplifica a construção de modelos, incorporando conceitos que facilitam a representação de aspectos relevantes do comportamento do sistema simulado. Como uma linguagem de programação, GASP IV concede ao analista um conjunto de comandos FORTRAN, elaborados para se obter as mais importantes funções na programação da simulação. Conceitos de modelagem são traduzidos pelo GASP IV em rotinas FORTRAN que podem facilmente ser usadas, permitindo

deste modo a ligação entre a modelagem e a programação, etapas importantes no sucesso de um estudo de simulação. Desta forma, simplifica-se grandemente o projeto e análise requeridos para solução do problema.

GASP IV possui cinco características que a tornam particularmente atrativa como uma linguagem de simulação:

1) É baseada em FORTRAN e não requer sistema de compilação separado. É facilmente mantida e pode ser implementada em computadores de diferentes fabricantes;

2) É modular e pode ser feita para se adaptar a todos os computadores que tenham compilador FORTRAN;

3) É fácil de aprender, haja vista que a linguagem FORTRAN é usualmente conhecida e somente conceitos de simulação necessitam ser dominados;

4) É usada para modelagem discreta, contínua ou combinada (contínua-discreta);

5) É facilmente modificada e estendida para encontrar as necessidades de aplicações particulares.

III.3.2. CONSIDERAÇÕES GERAIS

Numa simulação com GASP IV, as operações do sistema são consideradas a pontos discretos no tempo chamados "eventos". O comportamento do sistema é reproduzido pelo exame do sistema, aos tempos dos eventos. GASP IV assiste o analista, produzindo um tipo de modelagem e um conjunto de comandos de linguagem de programação que apressa e aperfeiçoa esta tarefa.

O objetivo de um modelo de simulação em GASP IV é reproduzir as atividades em que as entidades se inter-relacionam. Isto é feito definindo os estados do sistema e elaborando as atividades que movem o sistema de um estado a outro. Um siste

ma é dito estar num particular estado, quando todas as suas entidades estão em estados consoantes com aquele estado. As atividades, raramente, têm efeito instantâneo sobre o sistema. Elas tomam lugar em um período do tempo e, ao fim deste tempo, criam um novo estado do sistema. Eventos indicam o início e o fim das atividades. A Figura 2 abaixo, ilustra um simples atendimento, consistindo de um atendente e clientes que chegam para serem atendidos. As entidades, seus atributos e relações são mostradas também.

ENTIDADES	ATRIBUTOS	EVENTOS	ATRIBUTOS	RELAÇÕES
Clientes	Taxa de Chegada Tipo de Chegada Tipo de Serviço	Chegada de um Cliente	Tempo de Chegada Tipo de Cliente Prioridade de Serviço	Clientes que esperam na área de Atendimento Clientes que são atendidos Clientes que não são atendidos
Atendente	Taxa de Atendimento Tempo esperado para completar serviço Estado: Ocupado Ocioso	Fim do Serviço	Tempo de Conclusão Tipo de Cliente Prioridade de Serviço	

Figura 2 - Sistema de um Simples Atendimento

A operação do sistema pode ser descrita em termos de eventos do sistema. O sistema mostrado pode operar da seguinte maneira: um cliente (uma entidade) chega (um evento) em algum tempo específico (um atributo do cliente). Se o atendente (uma entidade) está livre (um atributo do atendente) um tempo de serviço é determinado baseado na distribuição de tempo de serviço do atendente (um atributo). Um cliente completa o serviço no tempo em que ele entra em serviço mais o seu tempo de serviço; o tempo de conclusão do serviço torna-se um atributo do cliente e do atendente. Um evento "fim de serviço" é programado para ocorrer ao tempo de conclusão. Se o atendente está ocupado quando o cliente chega, o cliente une-se à fila de espera (uma fila relatando os clientes esperando para serem atendidos) numa posição baseada na prioridade assinalada para os

clientes. A prioridade e o número de clientes na fila de espera são atributos da fila de espera.

Em GASP IV, as estruturas "entidade-atributo-relação", são estabelecidas pelas declarações de dados. Entidades são armazenadas como vetores e matrizes, cujos valores representam os atributos. O comando FORTRAN "DIMENSION SERVR (10,3)" pode representar 10 atendentes, cada um dos quais tem 3 atributos: (1) estado; (2) taxa de atendimento e (3) número de atendimentos realizados. SERVR (2,1) representa o estado do 2º atendente; SERVR (9,2) representa a taxa de atendimento do 9º atendente, etc.

III.3.3. FORMULAÇÃO DE MODELOS EM GASP IV

Entidades, atributos e relações fazem a estrutura estática de um modelo de simulação em GASP IV. Eles descrevem o estado do sistema, mas não como o sistema opera. Isto é tarefa dos eventos. A chave da simulação com GASP IV é a habilidade de organizar os eventos do sistema numa ordem tal, que eles são executados pelo computador numa ordem correspondente com a sua realização no mundo real.

Cada modelo de simulação em GASP IV consiste de:

- Um conjunto de programas dos eventos, que descrevem as regras de operação do sistema;
- Listas e matrizes que armazenam informações;
- Uma rotina executiva que dirige o fluxo de informação e controle dentro do modelo.

Cada programa de simulação em GASP IV é feito de subprogramas unidos por um programa executivo que organiza e controla sua execução.

GASP IV é organizado de modo a providenciar várias funções específicas, requeridas para cada simulação.

- Controle dos eventos;
- Matrizes de armazenagem de informações;
- Inicialização do estado do sistema;
- Coleção de dados para avaliação do sistema;
- Computação de estatísticas e geração de relatórios;
- Geração de variáveis aleatórias.

Estas funções podem ser apresentadas pela Figura 3.

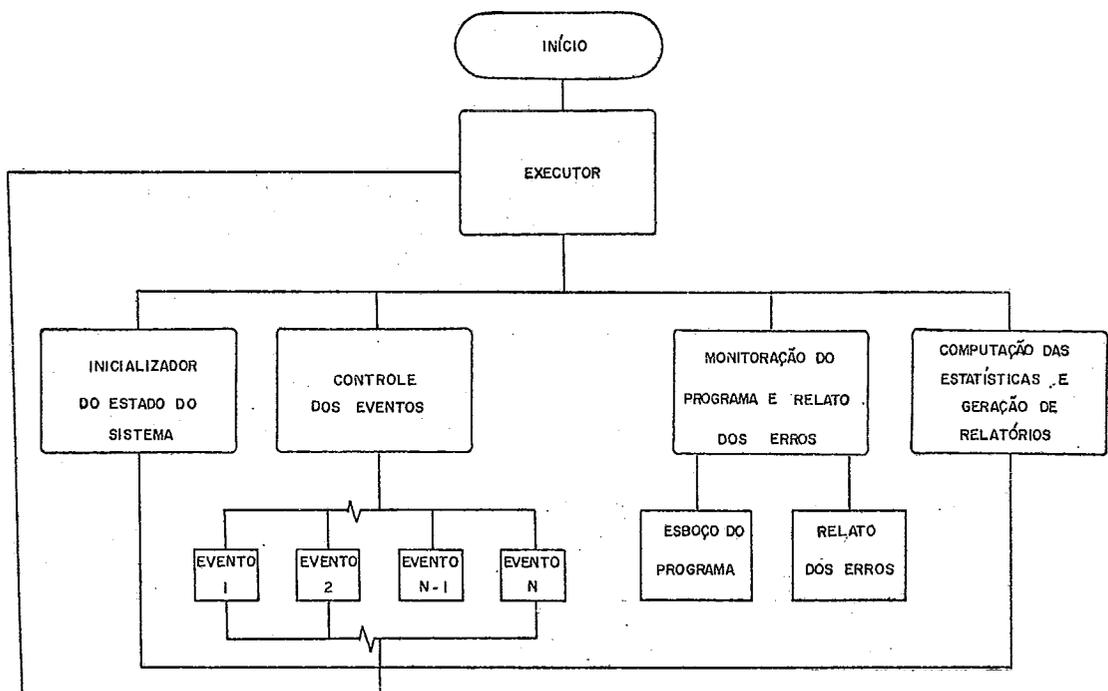


Figura 3 - Métodos Básicos de Controle do GASP IV

Um programa de simulação escrito em GASP IV consiste de duas partes: a parte do usuário e a parte do GASP IV. A parte do usuário contém: subprogramas para inicialização (programa principal e sub-rotina INTLC); equações para as variáveis de estado e condições definindo eventos-estados (sub-rotinas STATE e SCOND); definição do código dos eventos (sub-rotina EVNTS); sub-rotinas dos eventos (chamadas pela EVNTS); e procedimentos especiais para coleta de dados e impressão de resultados (sub-rotinas SSAVE, OPUT e UERR). A parte do GASP IV contém subprogramas que providenciam as seguintes funções: avanço do tempo e controle do programa (sub-rotina GASP), inicialização de dados e eventos (sub-rotinas DATIN, CLEAR e SET), armazenamento e recuperação de dados (sub-rotinas FILEM, RMOVE, CANCL, COPY e NFIND), coleta de dados, computação e impressão de relatórios (sub-rotinas COLCT, TIMST, TIMSA, HISTO, GPLOT, PRNTQ, PRNTS e SUMRY); monitoração do programa e impressão de erros (sub-rotinas MONTR e ERROR); geração de valores aleatórios (Funções DRAND, UNFRM, TRIAG, RNORM, RLOGN, ERLNG, GAMA, BETA, NPSSN, GAM) e suporte extra (funções SUMQ, PRODQ, GTABL e sub-rotina GDLAY).

III.4. XLSIM2/FOR

III.4.1. INTRODUÇÃO

A linguagem de simulação XLSIM2/FOR (Experimentation with Cellular Simulation Models 2 - versão FORTRAN) consiste na ampliação, através da inclusão de novas rotinas e versão para o FORTRAN, da linguagem XLSIM⁸, desenvolvida inicialmente como parte da pesquisa de doutorado de Ricardo Spinelli de Carvalho, sob a orientação de John G. Crookes, no Departamento de Pesquisa Operacional da Universidade de Lancaster, Inglaterra. A linguagem foi projetada como um instrumento de ensino de simulação de sistemas discretos em computadores digitais, enfatizando a aplicação da idéia de simulação celular (técnica de modelagem que agrupa atividades de um sistema

em células, de modo a aumentar a eficiência da experimentação com o modelo)^{8,9,10}. Entretanto, XLSIM incorpora outras facilidades tais como sequências antitéticas e amostras correlacionadas, que visam principalmente aumentar a eficiência da experimentação.

XLSIM2/FOR é composta por um conjunto de sub-rotinas FORTRAN que simplificam a articulação de conceitos e a execução de tarefas comuns em simulação, facilitando o trabalho de programação de modelos. Por outro lado a forma mnemônica de suas constantes, sub-rotinas e palavras reservadas contribui para que XLSIM2/FOR seja fácil de aprender e difícil de esquecer, preservando, dentro de suas limitações, a fluência característica da versão original.

A linguagem permite ao usuário, indiferentemente, a modelagem de simulação através das três fases sugeridas pelo prof. Tocher⁷ ou da organização por eventos que não reconhece explicitamente a existência de atividades condicionais.

III.4.2. CONSIDERAÇÕES GERAIS

O primeiro ponto a ser considerado é o acesso aos diversos elementos do modelo. O ideal é que tal acesso seja feito por um modo de referência próximo ao natural, com o mínimo de restrições impostas ao usuário. Em XLSIM2/FOR os diversos componentes do modelo (GRUPOS, CONJUNTOS, ENTIDADES, TRANSAÇÕES) recebem no momento de sua definição um nome, à escolha do usuário, pelo qual serão referenciados, enquanto que seus respectivos atributos serão acessados por sub-rotinas, cuja chamada é feita de forma quase coloquial: (O, A, EXIST). Assim, por exemplo, para que uma variável de nome LUGAR receba o valor igual ao tamanho de um conjunto ordenado chamado FILA, previamente definido, deve-se codificar:

LUGAR = O (TAMAN, FILA)

Outro requisito importante é a capacidade de efetuar operações aritméticas e executar sub-rotinas e funções de cálculo. Para as operações mais comuns (soma e subtração) com atributos dos elementos de um modelo, XLSIM2/FOR é dotada de sub-rotinas especiais (SOME, SUBTR) que executam diretamente tais operações, poupando a tarefa de um acesso indireto aos operandos. Para as demais operações numéricas, a execução se dará por meio de comandos ou sequência de comandos codificados na linguagem base (FORTRAN), sendo o valor resultante convenientemente armazenado através da sub-rotina SEJA. Para se dividir o valor de um atributo de nome SALDO, referente a uma entidade CONTA, por dois, poderia ser usada a seguinte sequência:

```
Z = 0 (SALDO, CONTA)/2
CALL SEJA (SALDO, CONTA, Z)
```

A geração de variáveis aleatórias, de grande utilidade em simulação, é obtida também por meio de rotinas (AMSTA e ALEAT). O atributo IDADE de uma transação denominada HOMEM pode ser gerado, a partir de uma distribuição POPUL previamente definida, fazendo, por exemplo:

```
NN = AMSTA (POPUL)
CALL SEJA (IDADE, HOMEM, NN)
```

Além destas facilidades, todos os cálculos relativos às estatísticas de interesse do usuário são efetuados automaticamente.

É muito comum também que os modelos de simulação impliquem na necessidade de operar conjuntos ordenados. Tais conjuntos são modificados constantemente, exigindo a localização, inserção e extração dos elementos que os compõem, bem como a reordenação em função de seus atributos ou alteração dos mesmos. Para tanto XLSIM2/FOR dispõe de facilidades especiais, em diversos graus de complexidade, a serem usadas de acordo com as necessidades do modelo. O tamanho de um conjunto, bem como os ocupantes de suas posições extremas, são informações de

acesso imediato, mantidas pela linguagem, respectivamente sob os nomes de TAMAN, PRMRO e ULTMO. Outras posições podem ser indicadas através de ponteiros, definidos pelo usuário e controlados por sub-rotinas próprias tais como APONT, AVNCE, RTCDA, CANCL. Assim, por exemplo, o segundo elemento de um conjunto ordenado de nome FILA pode ser identificado da seguinte forma, utilizando-se um ponteiro de nome DEDO previamente definido:

```
CALL APONT (DEDO, PRMRO, FILA)
CALL AVNCE (DEDO)
IDENT = 0 (ELMTO, DEDO)
```

Neste caso IDENT assumiria exatamente o valor do identificador do elemento apontado por DEDO, que será justamente o segundo do conjunto FILA.

As operações de entrada e saída de um conjunto são executadas respectivamente pelas sub-rotinas CLQUE e RTIRE, utilizando-se ou não de ponteiros. A sequência abaixo:

```
CALL RTIRE (PRMRO, FILAA, APROV)
CALL CLQUE (APROV, ULTMO, FILAB)
```

fará com que o primeiro elemento do conjunto FILAA seja dele extraído, temporariamente identificado como APROV e logo colocado em último lugar no conjunto FILAB. A própria linguagem se incumbirá de atualizar os valores dos atributos TAMAN, PRMRO e ULTMO de ambos os conjuntos. Para operações mais complexas, XLSIM2/FOR dispõe ainda de uma série de sub-rotinas (PCURE, JUNTE, ESVAZ, RMOVA, PSCNE, ORDNE, RTNJA) que podem eventualmente poupar trabalho ao usuário, embora não sejam de uso tão frequente como as anteriores.

Os modelos apresentam, por vezes, estruturas lógicas complexas que exigem da linguagem a capacidade de realizar testes e decisões que controlem eficientemente a sequência de operações a serem executadas. No caso de XLSIM2/FOR, a lógica do modelo é estruturada na própria linguagem base (FORTRAN) acrescida de algumas rotinas com funções específicas

cas (ETAPA, LEIA).

A alma de um modelo de simulação é, todavia, a forma pela qual seus diversos componentes interagem, modificando o estado do sistema. Isto pode ser feito basicamente de dois modos: por meio de eventos programados para ocorrerem em um dado instante, que são controlados necessariamente através de uma lista; por meio de atividades condicionais, o que exige que certas características do sistema sejam verificadas sempre que houver uma possibilidade de efetivação de mudanças no mesmo. Em XLSIM2/FOR é possível adotar uma ou outra alternativa, ou mesmo ambas, de forma combinada de acordo com o modelo de três fases sugerido por Tocher⁷. Para isto deverão ser usadas as rotinas EVNT e ETAPA, que controlam a evolução do modelo. Pela primeira são definidos os eventos que poderão ocorrer durante a simulação enquanto que, pela segunda, tem-se a possibilidade de avançar para o próximo evento previsto na lista interna mantida pela linguagem.

Sem dúvida alguma, o sucesso de uma simulação dependerá não apenas da eficiência com que o sistema é modificado, mas também da adequada conservação de informações relativas a este processo e de posteriores facilidades de acesso às mesmas. Tais informações podem, de forma geral, ser agrupadas em três categorias: instantâneas, séries temporais e estatísticas sumárias. A primeira engloba todas aquelas informações referentes ao estado atual do sistema, ignorando as etapas que o antecederam. O segundo tipo se refere àquelas informações que relacionam à cada observação do sistema a identificação do instante em que foi obtida, cardinal ou ordinalmente, formando um histórico do comportamento do sistema no tempo. A última espécie de informação é aquela dada por meio de médias, desvios, histogramas, valores extremos ou outros valores de interesse, resultantes de condensações e cálculos efetuados a partir de observações numéricas, de forma a evidenciar os principais aspectos nelas contidos.

As três categorias recebem de XLSIM2/FOR tratamento diverso. As informações do tipo instantâneo estão sempre dis-

poníveis ao usuário, cabendo-lhe porém a tarefa de selecioná-las e pesquisá-las individualmente. Este tipo de informação, no entanto, será normalmente de pouca utilidade no sentido, por hipótese, de obter dados relevantes para uma decisão entre alternativas. Seu maior valor é no sentido de ilustrar o funcionamento do modelo e secundariamente de auxiliar na depuração do mesmo. Para esta função, contudo, XLSIM2/FOR provê instrumento específico (CHAVE) mais eficiente.

Séries temporais podem ser trabalhadas através de históricos que são, essencialmente, listas definidas pelo usuário, quando delas necessitar, nas quais serão anotados os valores observados no decorrer da simulação e posteriormente obtidas as análises necessárias do ponto de vista de experimentação do modelo. A linguagem possui rotinas específicas para definição, anotação, análise e apresentação gráfica dos históricos.

Quanto às estatísticas, correspondentes ao terceiro tipo indicado de informações no modelo, uma vez tendo sido definidas pelo usuário, passam a ser trabalhadas pela linguagem, que se incumbe de efetuar as operações devidas, sempre que algum novo dado lhes é acrescentado. O acesso às estatísticas é facilitado por rotinas (HISTG, NAMST, MENOR, MAIOR, MEDIA, DSVIO) que tornam disponíveis as informações especificamente solicitadas e pela sub-rotina IMPMA que imprime todas as informações sobre uma estatística em formato padronizado.

Diversas rotinas adicionais relativas a aspectos de experimentação (análise de auto-correlação, variáveis antitéticas, etc) compõem a linguagem, estimulando o usuário a aprofundar a análise de seus modelos.

III.4.3. FORMULAÇÃO DE MODELOS EM XLSIM2/FOR

A sequência a ser seguida na codificação de um modelo em XLSIM2/FOR é aproximadamente a mesma desenvolvida para a formulação do modelo em si. O primeiro passo consiste na de finição das entidades, transações, grupos e conjuntos envolvidos, com seus respectivos atributos. As sub-rotinas indicadas para isto são DGRUP, DFINA, DFNA1, CRIE, CRIE1.

Em seguida, condicionadas aos resultados de interesse, devem ser definidas as estatísticas e/ou históricos que serão mantidos. As sub-rotinas STAT e HSTRC executam esta tarefa. Ainda nesta fase inicial devem ser identificados todos os eventos possíveis no modelo, através da sub-rotina EVNT, bem como as distribuições que serão usadas na programação dos mesmos, através da sub-rotina DSTRB. Com isso, as principais partes integrantes do modelo estão identificadas. As relações existentes entre todos estes elementos deverão então ser descritas nos blocos correspondentes a cada evento condicional. Nestes blocos é que se processarão realmente as modificações no sistema.

Como um evento provoca ou conduz a uma alteração no estado do sistema, quase sempre implicará na geração de uma nova informação que, sendo de interesse, será convenientemente guardada pela utilização das sub-rotinas ANOTE, RGTRE ou RGTBL, conforme o caso. Estas serão chamadas, em geral finalizando ou iniciando uma sequência de operações descritas por meio de comandos FORTRAN e sub-rotinas XLSIM de modificação do estado do sistema, tais como: SOME, SUBTR, CLQUE, RTIRE, DSTRU, APONT, etc.

Dentro dos blocos correspondentes e eventos ou atividades condicionais são indicados, através da sub-rotina PRGME, os eventos de ocorrência já prevista em um tempo futuro determinado e que passam a integrar a lista ordenada mantida pela linguagem. A consulta a esta lista, feita por meio da sub-rotina ETAPA, sempre que um bloco correspondente a um evento termina de ser executado, é que determinará o próximo passo a

ser executado, ou seja, para que bloco de comandos, correspondente a um novo evento, o programa deverá se deslocar.

Tendo sido codificadas todas as operações internas de cada bloco, tem-se completo o núcleo do modelo a ser simulado. A este se acrescentarão apenas:

a) uma lista de equivalência entre o nome dos atributos, constantes, etc., referenciados no modelo e seus respectivos números;

b) uma ou mais séries de comandos FORTRAN e sub-rotinas XLSIM2/FOR₁₈, relacionadas com aspectos do planejamento tático do modelo, tais como: condições iniciais, fim de simulação, auto-correlação, etc.,

c) comandos que controlem a execução da simulação para os diversos valores desejados das variáveis de decisão.

III.5. SIMSCRIPT II

III.5.1. INTRODUÇÃO

A linguagem SIMSCRIPT II é estruturada em função da descrição de sistemas envolvendo conceitos de "entidades", "atributos", "conjuntos", "estados" e "eventos". Uma "entidade" é uma classe de objetos descritos por uma coleção fixa de parâmetros chamados "atributos". "Conjuntos" são coleções de "entidades" individuais tendo certas propriedades comuns. O "estado" do modelo a um dado instante é completamente descrito pela lista individual das entidades, seus atributos e seus conjuntos. A dinâmica do sistema é representada pelas mudanças de estado, isto é, adição e deleção de entidades, mudança do valor dos atributos, dos conjuntos, ou combinação destes. Estas mudanças tomam lugar instantaneamente a pontos discretos no tempo simulado e são chamadas "eventos". O tempo ao qual um "evento" ocorre é definido num programa em SIMSCRIPT II

como o tempo corrente mais um incremento. Após a conclusão de qualquer evento, o tempo da simulação é automaticamente mudado para o tempo do próximo evento.

III.5.2. CONSIDERAÇÕES GERAIS

Como já foi dito, SIMSCRIPT II é baseado no conceito de que o estado do sistema pode ser descrito em termos de "entidades" (as coisas ou objetos que compõem o sistema), "atributos" (propriedades associadas com as entidades) e "conjuntos" (grupos de entidades). O usuário deverá especificar explicitamente todas as entidades com uma completa lista de seus atributos e possíveis conjuntos, na primeira etapa do desenvolvimento da simulação. Quando o usuário define uma entidade, ele está realmente definindo uma classe de objetos, cujos membros possuem uma mesma lista de atributos. Deste modo, por exemplo, parte de um sistema pode ser especificado desta forma:

ENTIDADE	HOMEM
ATRIBUTOS	PESO, ALTURA, IDADE
CONJUNTOS	SOBRAPO, SUCESSU

O modelo irá então operar sobre os valores dos atributos, bem como, sobre o conjunto de indivíduos homens que podem ser um ou vários. Nos problemas onde as entidades irão permanecer no sistema durante todo o tempo da simulação, as entidades são especificadas como "permanentes". Em outros casos, esta especificação não é possível por causa da tendência dos indivíduos de entrarem e saírem do sistema; por exemplo, pedidos a serem processados numa oficina, clientes em um supermercado, etc. Neste caso, a entidade é especificada como "temporária", e nenhuma restrição fixa é colocada sobre o número de indivíduos daquela entidade, os quais podem estar no sistema num certo instante qualquer. A qualquer tempo entidades temporárias individuais podem ser criadas ou destruídas.

Existe uma exceção ao conceito de atributos associados com entidades. No caso de entidades permanentes, não é necessária a associação de um atributo com uma entidade (o atributo é associado ao sistema como um todo) ou com duas entidades. A última opção tem o efeito de criação de uma matriz bi-dimensional. Um exemplo disto seria:

ENTIDADE	USINA
ENTIDADE	ALMOXARIFADO
ATRIBUTO	DISTÂNCIA

Dadas n usinas e m almoxarifados, um atributo de DISTÂNCIA estará associado com todo par (i,j) de usina e almoxarifado.

O estado do sistema em qualquer instante no tempo está plenamente descrito pelos valores dos atributos de todas as entidades individuais do sistema. Este estado é trocado pela ocorrência de um evento, o qual é representado por uma sub-rotina escrita com comandos SIMSCRIPT II quando

- Criamos ou destruimos uma entidade temporária individual;
- Trocamos os valores numéricos dos atributos.

Todas estas ações requerem a habilidade de localizar uma entidade individual e/ou atributo sobre o qual operar. A maior parte da linguagem SIMSCRIPT II é dirigida a prover métodos flexíveis e convenientes de executar as funções de armazenamento e recuperação de informações. Estes métodos servem para:

- Localizar uma entidade individual por seu nome (endereço); o qual tem permanecido disponível;
- Localizar uma entidade individual ou grupo de entidades pesquisando o conjunto. Esta pesquisa pode ser controlada dependendo dos valores dos atributos dos indivíduos no conjunto;

- Localizar uma entidade individual ou grupo de entidades por meio da pesquisa de todos os indivíduos da entidade.

Os atributos podem ser especificados por referência direta ou indireta à entidade individual com a qual eles estão associados. Eles podem estar associados com o sistema como um todo ao invés de estar apenas associado com uma entidade, como no caso do atributo tempo. Os valores dos atributos são modificados pelas expressões SIMSCRIPT II.

O procedimento para sequenciar as entidades individuais no conjunto é pré-especificado como LIFO, FIFO, ou ordenado num atributo específico. As entidades individuais são então enfileiradas ou removidas do conjunto de acordo com este procedimento. Nos casos especiais, entidades individuais podem ser removidas do conjunto pelo nome.

Um programa de simulação SIMSCRIPT II é composto de eventos bem definidos. As ocorrências separadas dos eventos são programadas no instante de tempo zero, para um tempo pré-especificado, sobre o controle do SIMSCRIPT II. Muitas ocorrências de eventos podem acontecer no mesmo tempo, cada ocorrência tendo sido previamente especificada pelo comando "CAUSE EVENT AT TIME X". É possível um dado evento reprogramar a si próprio e também é possível ser programada mais de uma ocorrência de um evento por outros eventos.

Os eventos são programados para qualquer tempo futuro e podem ser removidos da programação em qualquer tempo antes de serem executados. Em adição aos eventos "endógenos", os quais são programados dinamicamente pelo modelo, os eventos "exógenos" podem ser pré-programados no decorrer do processamento da simulação. Este último tipo de evento não está sujeito ao cancelamento ou reprogramação.

A saída é obtida através de uma classe especial de sub-rotinas as quais são compiladas por um gerador de relatórios. Os relatórios são usados para dar saída nas descrições de estado de um modelo em qualquer tempo, em qualquer grau de detalhe, utilizando formatos pré-estabelecidos, mas que

podem ser modificados. O conteúdo dos relatórios é especificado nos mesmos termos nos quais a estrutura estática do sistema foi definida (isto é, pelo uso dos atributos e nomes das entidades). Através do uso de comandos de repetição de linhas e colunas podem-se escrever matrizes uni ou bi-dimensionais de qualquer tamanho.

III.5.3. FORMULAÇÃO DE MODELOS EM SIMSCRIPT II

Cada programa de simulação em SIMSCRIPT II contém o comando:

START SIMULATION

que instrui o programa de controle do SIMSCRIPT a iniciar a simulação tirando instruções do mecanismo de controle do tempo simulado. Para isto, então, o programador deve inicializar o estado do sistema e providenciar eventos inicializadores que coloquem o sistema em movimento. Um típico programa de simulação em SIMSCRIPT é organizado desta forma:

MAIN

Declarações das matrizes de dados

Declarações e inicializações das entidades, atributos e conjuntos

Declarações e inicializações dos eventos

Especificação dos eventos externos

START SIMULATION

Comandos de controle

END

Quando o mecanismo de controle do tempo não encontra nada, isto é, não existe nenhum evento programado, o controle do programa passa para o comando colocado após o comando START SIMULATION e a simulação é terminada.

III.6. EXTENDED CONTROL AND SIMULATION LANGUAGE (ECSL)

III.6.1. INTRODUÇÃO

"Control and Simulation Language" foi originalmente concebida (pelo Grupo de Pesquisa Operacional da Companhia de Petróleo ESSO em 1960) como uma linguagem que procurava simplificar a descrição de um sistema complexo desenvolvido para simular as operações de carga e descarga de vários petroleiros da companhia. O objetivo da linguagem era puramente de comunicação e a tarefa de seu desenvolvimento foi dada ao grupo de programadores FORTRAN da companhia, que se mostrou insuficiente. Deste modo, a IBM providenciou um grupo auxiliar, sob a liderança de J. Buxton, que escreveu um programa para translação do CSL em FORTRAN. Esta linguagem, designada então por CLS1, foi usada por vários anos, tanto pela ESSO como por clientes da IBM, para a solução de uma ampla faixa de problemas.

Entretanto, com o passar do tempo algumas deficiências apareceram. Nova equipe da IBM foi formada e nova versão foi desenvolvida, com novas facilidades, sendo designada por CSL2. Pouco tempo depois, A.T. Clementson foi designado pela Honeywell para preparar um sistema para os computadores da série 200, o qual foi elaborado análogo ao CSL2, mas com outras facilidades. Esta versão ficou conhecida como ECSL (Extended CSL). ECSL foi planejado para ser usado num equipamento de cartões perfurados e desta forma, em muitas maneiras, também se assemelha com o FORTRAN. Comandos são perfurados cada um em um cartão (com continuações, se necessário) usando o mesmo conjunto de caracteres do FORTRAN. Contudo, ECSL é uma linguagem muito mais poderosa. Este poder extra é obtido de duas maneiras: primeiramente pela estrutura imposta ao programa; em segundo, pelos tipos adicionais de variáveis e comandos.

III.6.2. CONSIDERAÇÕES GERAIS

ECSL é principalmente estruturado em função de um tipo de variável chamada "entidade". Cada "entidade" é, puramente, uma representação de um objeto no sistema sendo modelado. "Entidades" similares são arranjadas em "grupos" ou "classes". Por exemplo, um modelo de um porto conterá uma "classe" de navios, uma "classe" de ancoradouros, etc. O segundo mais importante tipo de variável é o "conjunto" (ou "fila"). Representa uma relação (lista) de "entidades" de uma "classe" particular. Cada "entidade" pode ter um número de "atributos". Estes atributos podem ser do seguinte tipo: "integer", "real", "boolean", "string", "time" ou "histogram". Variáveis independentes deste mesmo tipo podem também ser definidas - elas podem ser vistas como "atributos" do sistema. Os atributos "time" são de considerável importância; eles permitem a seleção de cada um dos tempos que necessitam ser considerados durante a corrida da simulação. Cada "time" representa a quantidade de tempo que ainda tem que passar antes de a "entidade" ter completado sua atividade corrente. O sistema monitor, automaticamente, atualiza estas células de "time", assim como move o CLOCK para frente.

ECSL tem uma ampla variedade de tipos de comandos. Isto corresponde ao fato de que os tipos de condição e ação, necessários na simulação, são mais variados que aqueles requeridos num trabalho numérico. São os seguintes: definições, designações, comparações, manipulações de conjuntos, comandos esttísticos, entrada/saída, comandos estruturais, comandos de controle, comandos de combinação, comandos de dados.

A "visão global" do ECSL é que qualquer sistema consiste de "entidades" que se correlacionam de tempo em tempo, para realizarem trocas no estado do sistema. Um programa em ECSL consiste na descrição de todos os tipos de atividades que podem ocorrer no sistema que está sendo simulado.

O sistema é descrito em termos de classes de entidades, conjunto de entidades, e matrizes de números (dados). Em

geral, as dimensões das classes, conjuntos e matrizes são pré-especificadas. Um exemplo deste tipo de especificação é:

```

CLASS      4 CRANE
CLASS      4 CRANE SET LOADING AVAILABLE OUTOFSERVICE
ARRAY     TRUCKWT (50)

```

Em todos os casos, os números indicam as dimensões associadas com a classe, matriz ou conjunto. Matrizes podem também ser bi ou tri-dimensionais. Entidades, as quais são membros individuais de uma classe, são normalmente referidas ' pelo subscrito da classe. O subscrito pode também ser especificado indiretamente.

O estado do sistema, usando ECSL, será dado por listas de entidades pertencendo a cada conjunto no sistema, e por dados numéricos (geralmente associados com entidades) armazenados na matriz. Os dados da matriz relativos às entidades individuais são referenciados por subscritos. Os atributos de cada entidade são colocados num vetor de 0,1 ou 2 dimensões e são referidos pela posição no vetor. Isto facilita a associação de um grande número de atributos com as entidades de uma classe.

O estado de um sistema pode ser trocado:

- Adicionando ou deletando entidades nos conjuntos;
- Trocando o valor numérico do dado armazenado em uma ou mais posições de um "array".

Uma entidade pode ser incluída no início ou no fim de um dado conjunto; pode pertencer a qualquer número de conjuntos em um mesmo tempo; e as entidades no conjunto podem ser ordenadas com qualquer critério em qualquer tempo.

Alterações podem ser feitas por meio da "álgebra dos conjuntos ordenados", os quais permitem extensas modificações com comandos relativamente simples.

ECSL fornece uma variedade de procedimentos de execução de testes com complexos relacionamentos lógicos. O resultado final de cada teste é a seleção de um dos dois destinos: sucesso ou fracasso.

Os testes podem ser baseados em simples relacionamentos aritméticos. Eles também podem ser baseados em certas condições.

Complexos testes são possíveis, nos quais qualquer número de comandos podem estar relacionados. Estes testes em cadeia podem ser aplicados diretamente a todos os membros de um conjunto sob o controle de comandos apropriados.

As expressões aritméticas operam diretamente sobre os atributos específicos ou posições individuais das matrizes para executar a transformação dos dados numéricos. ECSL também fornece uma geração de números randômicos para facilitar a introdução flexível de variáveis randômicas independentes.

Um programa ECSL é definido como um programa dividido em setores referidos como atividades. As variáveis podem ser definidas no local para cada atividade, e transferências entre atividades são permitidas.

Em qualquer ponto no tempo, os valores correntes das variáveis aritméticas especificadas, matrizes ou histogramas podem ser escritos. Histogramas são essencialmente tabelas nas quais são escritas distribuições de frequência dos dados coletados durante o curso da simulação.

III.6.3. FORMULAÇÃO DE MODELOS EM ECSL

A estrutura de um programa típico em ECSL pode mais facilmente ser descrito pela orientação da atividade: de fora (vindo) para (em direção) o interior. O programa é composto de 5 (cinco) partes principais: definições, inicialização, atividades, finalização e dados.

A seção "definições" introduz ao tradutor aqueles nomes que serão usados para descrever as variáveis. Estas variáveis

veis serão utilizadas para relatar o estado do sistema; desta forma, as definições realmente especificam a dimensão do sistema.

A seção "inicialização" é usada para estabelecer os valores iniciais das variáveis, isto é, ela especifica o estado inicial do sistema.

A seção "atividades" especifica o comportamento dinâmico do sistema. Esta seção consiste de um número de subseções, cada uma das quais intervem com um tipo particular de troca. Cada subseção é conhecida como uma "atividade" e contém comandos que checam a condição no sistema para decidir se a troca deve ser feita e comandos para efetuar a troca.

A seção "finalização" usualmente diz respeito à produção da impressão de resultados do comportamento do modelo durante a corrida. Estas instruções são submetidas somente após o fim do período da simulação.

A seção "dados" especifica valores iniciais para algumas variáveis e é usada em conjunto com a seção "inicialização".

O compilador ECSL então integra estas cinco seções em um programa completo, de uma maneira óbvia e simples. Primeiramente, a seção "inicialização" é executada. Então o primeiro tempo expressivo (da simulação considerada) é determinado pelo sistema. Cada uma das atividades é verificada neste tempo: onde as condições especificadas são encontradas para existir, as ações consequentes são executadas; caso contrário, nenhuma ação é tomada. Este processo é repetido para todos os tempos expressivos até o tempo final dado pelo comando "ACTIVITIES". Então a rotina "finalização" é executada.

Toda a estrutura do ECSL até aqui relatada, pode ser resumida pelo diagrama da figura abaixo:

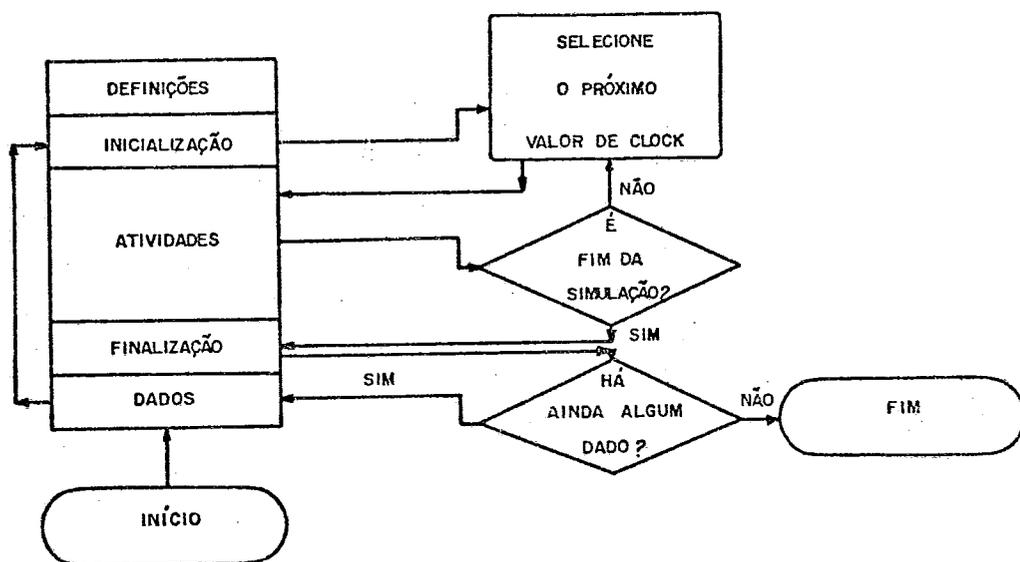


Figura 4 - Estrutura de um Programa em ECSL

III.7. AS LINGUAGENS E ALGUMAS DE SUAS FUNÇÕES

Apresentaremos, a seguir, um quadro geral de algumas funções das linguagens apresentadas no tópico anterior, funções estas relativas a alguns procedimentos do capítulo II. Convém lembrar que não é um quadro exaustivo, mas apenas uma breve indicação destes procedimentos. Também não é nossa intenção tecer maiores detalhes sobre eles, haja vista que os mesmos encontram-se bem descritos nos respectivos manuais e/ou livros já citados.

PROCEDIMENTO	ATIVIDADE	LINGUAGENS E/OU PROGRAMAS				ECSL (COMANDOS)
		SIMSRIPT II (COMANDOS)	GASP IV (SUBROTINAS E/OU FUNÇÕES)	GPSS (BLOCOS)	XLSIMZ/FOR (SUBROTINAS E/OU FUNÇÕES)	
VALIDAÇÃO	DEPURAÇÃO LÓGICA	BEFORE AFTER TRACE	MONTR ERROR UERR	TRACE UNTRACE	CHAVE	SWITCH CHECK SWITCH CHECK ON SWITCH CHECK OFF SWITCH CHECK ON AFTER
	IMPRESSÃO DE RESULTADOS	ACCUMULATE COMPUTE TALLY	SUMRY PRNTQ GPLOT HISTO PRNTS	TABLE PRINT REPORT GRAPH	IMPWA GRFCO	PICTURE
	TESTES ESTATÍSTICOS	?	?	?	KOLMO	?
PLANEJAMENTO TÁTICO	CONDIÇÕES INICIAIS	?	GPLOT CLEAR	CLEAR RESET GRAPH	HSTRC LIMPE ANOTE GRFCO	SWITCH ADD OFF SWITCH ADD ON SWITCH ADD ON AFTER
	REDUÇÃO DE VARIÂNCIA	?	?	?	ANET WATET RENCZ MUDE	?
	AUTOCORRELAÇÃO	?	?	?	CORGM CRBLC	?
EXPERIMENTAÇÃO		START SIMULATION BETWEEN V. END OF SIMULATION	GASP DATIN SET INTLC	START TERMINATE SIMULATE	CMECE RCMCE ATIVA DTIVE	RESTART CELL RECYCLE

Quadro 2 - Funções Usuais das Linguagens Consideradas

Pode ser observado pelo quadro 2, apesar de não ser completo, que todas as linguagens possuem, de uma forma ou de outra, umas mais outras menos, procedimentos que, de alguma maneira permitem a realização das tarefas de 5 a 8, mencionadas no capítulo II, de um projeto de simulação. Entretanto, em todos os casos há a necessidade de se combinar estas funções, condicionando-as à execução de determinado objetivo. Por exemplo, caso queiramos, na fase de depuração de um programa em SIMSCRIPT II, checar os subscritos para as operações FILE e REMOVE num certo conjunto QUEUE, utilizando para isto o comando BEFORE, devemos codificar o seguinte conjunto de informações:

Na seção "preâmbulo":

```
BEFORE    FILING AND REMOVING FROM QUEUE, CALL CHECK
```

Na própria rotina:

```
ROUTINE CHECK GIVEN ENTITY, SUB1 AND SUB2
```

```
DEFINE ENTITY, SUB1 AND SUB2 AS INTEGER VARIABLES
```

```
IF 0 < SUB1 ≤ M AND 0 < SUB2 ≤ N, RETURN
```

```
OTHERWISE...
```

```
    PRINT 2 LINES WITH SUB1 AND SUB2 THUS
```

```
    INCORRECT DIMENSIONS IN SET QUEUE
```

```
        SUB1 = * * * *      SUB2 = * * * *
```

```
CALL DUMP
```

```
STOP
```

```
END
```

Por outro lado, caso estejamos interessados na determinação do período transiente, utilizando uma das regras práticas sugeridas por Flávio Faria¹⁹, devemos utilizar, entre outros, os seguintes comandos; em XLSIM2/FOR:

```

C
C   DEFINIÇÃO DOS ELEMENTOS DO MODELO
C
. . . . .
CALL STATC (ESTAT1, NINTG, 0, 10, 5, 5)
. . . . .
CALL EVNTØ (PAUSA, 0, GLBAL, 0, 0)
. . . . .
CALL HSTRC (ANØTA, 1, 1000)
. . . . .

```

```

C
C   CONDIÇÕES INICIAIS
C
. . . . .
CALL PRGME (PAUSA, 50)
. . . . .

```

```

C
C   EVENTO PAUSA NA EXPERIMENTAÇÃO
C
. . . . .

IX1 = MÉDIA (ESTAT1)
CALL ANØTE (IX1, 1, ANØTA)
CALL LIMPE (ESTAT1)
CALL PRGME (PAUSA, 50)
. . . . .

```

```

C
C   EVENTO FIM DE SIMULAÇÃO
C
. . . . .
CALL GRFCØ (ANØTA, 1)
. . . . .

```

Em geral, conforme pode ser observado, esta preparação é trabalhosa e, algumas vezes, inviável em determinadas linguagens, bem como será repetida, na maioria das vezes, em todos os projetos de simulação que se desenvolva, mudando-se apenas os nomes e mnemônicos utilizados. Desta forma, costuma-se, já que as fases de formulação do modelo, preparação de dados e tradução do modelo são, em sua maioria (principalmente em problemas complexos de situações reais), bastante trabalhosas, relegar a um segundo plano estas etapas do estudo. Isto pode ser observado em todos os trabalhos^{31,32,33,34} até hoje desenvolvidos pelo autor desta.

Por outro lado, já há algum tempo Conway¹⁸ comentava: "em virtude disto, os resultados alcançados podem ser decepcionantes em relação ao esforço despendido".

Procurando minimizar este problema, nossa proposta, apresentada no capítulo seguinte, procura colocar à disposição do analista de simulação, um conjunto de procedimentos, que, utilizados de forma interativa, poderão permitir a execução das tarefas de 5 a 8 de uma forma agradável e menos exaustiva, tornando possível o desenvolvimento global do estudo e a obtenção de resultados mais confiáveis.

III.8. A ESCOLHA DA LINGUAGEM XLSIM2/FOR

A escolha da linguagem XLSIM2/FOR como apoio ao sistema desenvolvido e a ser apresentado no capítulo seguinte deveu-se a diversos fatores, dentre os quais podemos citar:

- um melhor conhecimento sobre a linguagem do que sobre as demais;

- um melhor suporte sobre a mesma, tendo em vista que o orientador deste trabalho é o próprio autor dela;

- ela está implantada no centro de computação da empresa onde este trabalho foi desenvolvido;

- ela possui um maior número de procedimentos que as demais, já implantados, e que facilitarão o desenvolvimento do sistema;

- ela já faz parte de um projeto maior, que é tornar a simulação um procedimento interativo;

- as próprias facilidades e vantagens que a linguagem apresenta ao desenvolvimento de simulação discreta.

Entretanto, convem lembrar mais uma vez, que o sistema poderá ser implementado com qualquer outra linguagem, desde que se desenvolvam os mesmos procedimentos dentro da linguagem escolhida.

CAPÍTULO IV

A SUGESTÃO: O SISTEMA ANALISADOR

IV.1. DELINEAMENTO DO SISTEMA

Analisando as atividades de desenvolvimento de um estudo de simulação, citadas no capítulo II, em especial as de 5 a 8, observa-se que existe uma estrutura em sua utilização. Apesar de não ser uma regra geral, normalmente iniciamos com a depuração lógica, que exige atividades de correr, dar um "trace" no programa, corrigir prováveis erros, correr, etc.

Após esta fase, passamos à parte do planejamento tático, que envolve as atividades de:

- determinação do período transiente: definição de histórico, correr, anotação, limpeza de estatística, plotação de gráficos, etc.

- determinação do efeito de auto-correlação: correr, fazer correlograma e/ou correlobloco, etc.

- determinação do tamanho da corrida: correr, impressão de resultados, etc.

Logo após passamos à validação, envolvendo atividades de correr, impressão de resultados, testes de aderência (Kolmogorov-Smirnov, Qui-quadrado, etc.) etc.

Finalmente, chegamos a experimentação com as seguintes atividades: correr, guardar o estado atual da corrida, restabelecer o estado inicial da corrida, modificar conjuntos, alterar atributos de elementos, reprogramar eventos, trocar parâmetros de distribuições, etc.

Considerando que esta estrutura pode não ser a única, procuramos apresentar um conjunto de funções colocadas de forma individual, de modo que o usuário possa usá-las como melhor lhe convier.

Por outro lado, nossa proposta também não pretende apresentar a melhor maneira de se realizar estas atividades, mas sim um conjunto criterioso de procedimentos que possam automatizá-las. Deste modo, achamos que estamos dando mais uma contribuição para tornar menor a defasagem entre a formulação do problema e a interpretação dos resultados.

IV.2. O SISTEMA ANALISADOR

Considerando que os recentes desenvolvimentos da técnica de simulação discreta em computador digital são os geradores interativos^{14,15,21}, nossa proposta não podia deixar de seguir o mesmo enfoque: a análise interativa dos resultados de um estudo de simulação. Serã mostrada e foi elaborada para ser utilizada com a linguagem XLSIM2/FOR. Entretanto, queremos crer que a sugestão é geral, podendo ser implantada com qualquer linguagem, desde que se façam os mesmos desenvolvimentos' nesta determinada linguagem.

Nosso sistema pretende, após elaborado o modelo e feita a sua tradução para a linguagem desejada, no caso XLSIM2/FOR, permitir ao usuário o desenvolvimento, de forma conversacional, das seguintes tarefas que permitam uma análise dos resultados do modelo:

1) - CORRER O MODELO: - serve para correr o modelo desenvolvido durante "X" unidades de tempo adicionais, podendo ser usada a qualquer instante, bastando digitar-se a palavra "CORRER" e informando a seguir a quantidade de tempo que se deseja simular.

2) - LIGAR CHAVES OU COMENTÁRIOS: - este procedimento serve para:

a) ajudar o usuário na depuração de seu modelo de simulação; deve-se digitar a palavra "LIGAR", seguida, também, da palavra "CHAVES", dando a seguir o número da chave que se quer ligar (vide tabela na página 98 do manual do XLSIM2/FOR²³

- caso se queira parar, basta digitar 0 e caso se queira ligar todas as chaves de uma só vez, deve-se digitar o número 8); se a chave está ligada, todas as sub-rotinas XLSIM2/FOR por ela monitoradas, quando utilizadas pelo usuário, têm seus nomes e respectivos argumentos mostrados na tela do terminal ou no formulário da impressora.

b) gerar a impressão dos comentários existentes após cada pergunta do sistema analisador, permitindo maiores esclarecimentos sobre a mesma; para isto, após digitar-se a palavra "LIGAR", digita-se a palavra "COMENTÁRIOS".

3) - DESLIGAR CHAVES OU COMENTÁRIOS: - este procedimento anula as funções do procedimento anterior; deve-se então, após codificar a palavra "DESLIGAR", seguir as mesmas etapas da função anterior.

4) - TIRAR UMA FOTOGRAFIA DAS CONDIÇÕES ATUAIS DA CORRIDA: - geralmente a experimentação com um modelo de simulação compreende um certo número de corridas. No início da corrida, ou assim que a corrida atinja um ponto no tempo simulado, que deva ser considerado como condição inicial para todas as corridas subsequentes, este procedimento permite o registro destas condições. Basta, para tanto, digitar a palavra "TIRAR".

5) - LIMPEZA DE ESTATÍSTICAS E/OU HISTÓRICOS: - às vezes as informações contidas numa estatística ou histórico tornam-se inúteis e o usuário necessita limpá-los para reutilizá-los. Para obter esta informação, o usuário após digitar a palavra "LIMPEZA", deve informar a seguir a ordem de definição da estatística e/ou histórico, conforme o caso. Para interromper o procedimento deve digitar o número 0.

6) - RELATÓRIO (IMPRESSÃO) DE ESTATÍSTICAS E/OU GRÁFICOS: - este procedimento deve ser usado sempre que o usuário estiver interessado em imprimir o conjunto completo de informações, inclusive o histograma associado a uma estatística específica, ou as séries de dados anotados em histórico, na forma de gráficos. Para isto deve digitar a palavra "RELATÓRIO", se-

guida da ordem de criação da estatística ou histórico, e de, somente no caso de gráfico, um valor inteiro menor ou igual a 130, para indicar um limitante superior para o agrupamento máximo desejado pelo usuário, na execução do gráfico.

7) - AUTO-CORRELAÇÃO (CORRELOGRAMAS E/OU CORRELOBLOCOS): - permite a análise de auto-correlação nas séries de dados anotados em históricos previamente definidos; a primeira análise permite a execução de um correlograma e a segunda, de um correlobloco: para tanto, deve-se digitar a palavra "AUTO-CORRELAÇÃO", seguida de "SIM", número da série a ser considerada, ordem de criação do histórico, e o valor máximo do lag para o qual deve ser estimado o coeficiente de auto-correlação; para o caso de correlobloco, segue-se procedimento idêntico, sendo diferente apenas o último parâmetro, que neste caso é o número máximo de observações por bloco.

8) - RESTABELECER AS CONDIÇÕES INICIAIS DESTA CORRIDA: - ao final de uma corrida, algumas alterações são feitas no modelo, antes que a próxima corrida seja executada, dependendo do tipo de experimentação que-está sendo feita; este procedimento permite voltar a corrida às mesmas condições quando do uso do procedimento "TIRAR UMA FOTOGRAFIA DAS CONDIÇÕES ATUAIS DA CORRIDA". Para tanto, basta digitar-se a palavra "RESTABELECER". Este procedimento só deve ser usado após uso do procedimento nº 4.

9) - TESTES DE KOLMOGOROV-SMINORV: - este procedimento permite a execução do teste de aderência do tipo Kolmogorov-Smirnov^{1,17}; deve-se, após digitar a palavra "TESTES", informar a ordem de definição da estatística a ser comparada, informando a seguir, uma a uma, a frequência da distribuição de que se quer testar a aderência (para parar deve-se digitar 0).

10) - ALTERAR ATRIBUTO DE ENTIDADE E/OU TRANSAÇÃO: - caso se necessite alterar o atributo de uma entidade ou transação do modelo, para efeito de experimentação, deve-se digitar, após a palavra "ALTERAR", a ordem de definição ou criação da entidade ou transação, a ordem de definição do atributo a ser alterado, bem como o seu novo valor.

11) - MODIFICAR CONJUNTOS (RETIRAR OU COLOCAR ELEMENTOS): - ainda no caso de experimentação, necessita-se às vezes, retirar ou colocar elementos em conjuntos previamente definidos; deve-se, após, digitar a palavra "MODIFICAR", informar a ordem de definição do conjunto, o tipo de modificação (RETIRAR OU COLOCAR): - caso seja, "RETIRAR", informar, a seguir, que elemento (PRIMEIRO, ÚLTIMO OU QUALQUER); no caso de "QUALQUER", informar ainda o tipo de elemento (ENTIDADE OU TRANSAÇÃO) e a ordem de definição do mesmo; - caso seja "COLOCAR", informar o tipo de elemento (ENTIDADE OU TRANSAÇÃO), sua ordem de definição e a posição desejada (PRIMEIRO OU ÚLTIMO).

12) - REPROGRAMAR EVENTOS (PROGRAMAR NOVOS OU APAGAR EVENTOS JÁ PROGRAMADOS): - por vezes, antes que nova corrida comece, devemos apagar ou programar novos eventos. Este procedimento permite esta reprogramação: - deve-se digitar, após a palavra "REPROGRAMAR", o tipo de reprogramação desejada (PROGRAMAR OU APAGAR), a ordem de definição do evento, o intervalo de tempo para a ocorrência do evento (no caso de APAGAR esta informação não é necessária) e as transferências' (tipo - ENTIDADE OU TRANSAÇÃO - e ordem de definição) envolvidas no evento.

13) - ATIVAR CÉLULAS: - quando se estiver usando estrutura celular, na hora da experimentação é necessário ativar determinadas células; este procedimento permite esta operação, bastando digitar após a palavra "ATIVAR" o número da célula que se quer ativar (para parar indica-se 0)

14) - DESATIVAR CÉLULAS: - pelo mesmo motivo citado no procedimento anterior, em determinadas ocasiões é necessário desativar as células ativas; este procedimento permite isto, bastando digitar-se a palavra "DESATIVAR".

15) - TROCAR PARÂMETROS DE DISTRIBUIÇÕES: - ao final de uma corrida, algumas alterações são feitas no sistema antes que a próxima corrida seja executada, dependendo do tipo de experimentação que está sendo feita. Uma destas alterações pode ser relativa a um padrão de chegadas, taxas de atendimento ou qualquer outra alteração de parâmetro de distribuição; este procedimento permite esta função: deve-se digitar a palavra "TROCAR", a ordem de definição da distribuição, o tipo da distribuição (NORMAL, ERLANG, UNIFORME OU EMPÍRICA) e os parâmetros respectivos (por exemplo, no caso de NORMAL, média e desvio padrão).

16) - ANOTAR, EM HISTÓRICOS, PARÂMETROS (MÉDIA, MÁXIMO, MÍNIMO, DESVIO PADRÃO, TAMANHO) DAS ESTATÍSTICAS: - quando se deseja definir o período necessário à eliminação do efeito de desequilíbrios gerados no início de operação do modelo, uma das alternativas é observar a evolução da média, máximo, mínimo, etc. das estatísticas mantidas. Para isto, deve-se executar, a intervalos convenientes, a anotação destes parâmetros em históricos previamente definidos, para posterior plotação em gráficos, permitindo deste modo determinar a situação de regime estacionário¹⁹; esta função permite executar esta atividade; para tanto, após digitar-se a palavra "ANOTAR", deve-se informar pela ordem:

- o parâmetro que se deseja anotar (média, máximo, etc.);
- a ordem de definição da estatística;
- a ordem de definição do histórico;
- a série do histórico que será usada.

17) - DEFINIR HISTÓRICOS: - para utilização do procedimento anterior, bem como para análises de auto-correlação, é necessária a definição de históricos que serão usados para guardar resultados gerados pelo modelo; para isto, digita-se a palavra "DEFINIR", informando a seguir o número de séries que serão usadas, bem como o número máximo de valores destas séries.

18) - MUDAR A SEQUÊNCIA DE VALORES ALEATÓRIOS: - este procedimento permite a mudança da semente de uma sequência de números ou variáveis aleatórias, simplificando o controle sobre esta sequência; digita-se a palavra "MUDAR", informando a seguir se é "SEQUÊNCIA ALEATÓRIA" ou "DISTRIBUIÇÃO", qual a ordem de definição da mesma e o novo valor da semente.

19) - REINICIALIZAR A SEMENTE DA SEQUÊNCIA DE VALORES ALEATÓRIOS: - este procedimento realimenta a sequência aleatória com a mesma semente inicialmente definida para ela ou a modificada pelo procedimento anterior; deve-se informar "REINICIALIZAR", seguida de "SEQUÊNCIA ALEATÓRIA" ou "DISTRIBUIÇÃO", conforme o caso, bem como a ordem da mesma.

20) - GERAR VARIÁVEIS ANTITÉTICAS: - dentre as possíveis técnicas de redução de variância, o uso de variáveis antitéticas apresenta-se de fácil implementação; este procedimento permite esta facilidade, bastando digitar a palavra "GERAR", seguida de "SEQUÊNCIA ALEATÓRIA" ou "DISTRIBUIÇÃO" e da ordem de definição da mesma.

21) - CANCELAR A GERAÇÃO DE VARIÁVEIS ANTITÉTICAS: - cancela a função do procedimento anterior; deve-se digitar a palavra "CANCELAR" seguida das mesmas informações anteriores.

22) - PROCURAR ELEMENTOS EM CONJUNTOS OU GRUPOS: - em determinados experimentos, há a necessidade de pesquisa de certos conjuntos ou grupos, no intuito de selecionar determinado elemento para alteração de alguns de seus parâmetros; este procedimento permite esta facilidade. Para tanto, deve-se

digitar, após a palavra "PROCURAR", as seguintes informações, pela ordem:

- o local (grupo ou conjunto) em que se encontra o elemento;

- a ordem de definição do grupo ou conjunto;

- o número do atributo pelo qual o elemento será selecionado (caso a pesquisa seja num grupo, o uso do atributo tamanho é permitido);

- o valor pelo qual o elemento será selecionado (é permitido o uso de máximo ou mínimo).

23) - LISTAR VALOR(ES) DE ATRIBUTO(S) DE ELEMENTO(S): - ainda com relação à experimentação, em determinados instantes pode-se necessitar saber o(s) valor(es) de certo(s) atributo(s) de determinado(s) elemento(s) do modelo; o uso deste procedimento permite esta facilidade. Para tanto, deve-se digitar, após a palavra "LISTAR", as seguintes informações, pela ordem:

- o tipo de elemento (entidade ou transação);

- a ordem de definição do elemento;

- a ordem de definição do atributo.

24) - PERMUTAR POSIÇÃO DE ELEMENTO DENTRO DE CONJUNTO: - em determinadas simulações existe a necessidade de se alterar a posição de determinado elemento dentro de um conjunto. Este procedimento permite a realização desta tarefa; para tanto, deve-se digitar, após a palavra "PERMUTAR", as seguintes informações, pela ordem:

- a ordem de definição do conjunto;

- o tipo de elemento a ser permutado;

- a ordem de definição deste elemento;

- a ordem de definição do atributo do elemento a ser permutado.

25) - ORDENAR CONJUNTO(S): - em algumas situações, antes de se alterar a posição de um determinado elemento dentro de um conjunto, necessita-se ordenar este conjunto. O presente procedimento permite esta facilidade; para tanto, após digitar-se a palavra "ORDENAR", deve-se informar, pela ordem, o seguinte:

- a ordem de definição do conjunto;
- a ordem de reagrupamento desejada (crescente ou decrescente);
- a ordem da definição do atributo, pelo qual o conjunto será reagrupado.

26) - DESTRUIR ELEMENTO(S) CRIADO(S): - em determinados instantes da simulação, quando se está fazendo experimentação, certas transações podem tornar-se desnecessárias; neste caso é conveniente destruí-las, ocasionando com isto liberalização de memória usada para guardar suas informações. Este procedimento permite a realização desta tarefa; para tanto, após digitar-se a palavra "DESTRUIR", deve-se informar a ordem de definição da transação a ser destruída.

27) - FINALIZAR A ANÁLISE: - permite encerrar a utilização do sistema analisador; para isto, basta digitar a palavra "FINALIZAR".

As instruções até aqui relacionadas são colocadas, com comentários adicionais, à disposição do usuário, quando o sistema estiver sendo utilizado. Para isto, deve-se, no início do programa representativo do modelo, digitar o comando seguinte:

```
CALL INCLZ(1)
```

O argumento 1 indica que o sistema analisador será usado no decorrer da simulação. Caso não se deseje usar tal sistema, o usuário deverá digitar então o seguinte comando:

```
CALL INCLZ(0)
```

Tendo em vista que foi desenvolvido para ser utilizado com a linguagem XLSIM2/FOR, o manual²³ desta linguagem contém informações adicionais, e deve ser usado sempre que possível, para melhor uso do sistema.

IV.3. ALGUNS EXEMPLOS DE UTILIZAÇÃO DO SISTEMA ANALISADOR

IV.3.1. DEPURAÇÃO

Uma das grandes dificuldades iniciais, após a formulação do modelo e sua codificação numa determinada linguagem é a depuração lógica do programa. É uma incessante sequência de: correr, erro, correção, correr, erro, correção, etc. Os procedimentos 1, 2, 3, 4 e 9, bem como a inclusão do comando CALL INCLZ(1) no programa, permitem esta sequência de uma forma fácil e ágil. Suponhamos, então, que já tenhamos o modelo codificado (conforme listagem das páginas 63 e 64), compilado e linkeditado, pronto para correr. Iniciamos e submetemos o programa. O sistema será colocado à disposição do usuário: este deverá usar a opção "CORRER O MODELO" para 0 unidades de tempo. Isto deve ser feito, para permitir, caso não haja erro na inicialização, o uso do procedimento "TIRAR UMA FOTOGRAFIA DAS CONDIÇÕES ATUAIS DA CORRIDA" que nos proporcionará, caso algum erro ocorra, restabelecermos estas condições e apurarmos o erro. Isto será mostrado mais adiante. No entanto, isto não acontece, conforme pode ser visto na página 65.

REAL MEDIA

LOGICAL EXIST, LEIA, RMOVA

INTEGER O, A, AMSTA

INTEGER YY(3000), VETYY(10, 50), CONTTY(10)

INTEGER TRNSF(20), CHAVE(7), VALOR(40), CELLA(50), FREQC(40),

*ENTDD, TRSAO, CONJT, PONTR, PRMRO, ULTMO, PDSOR, SUCSR, ELMTO,

*TAMAN, ENTRA, SAIDA, AGORA, ERLNG, UNFRM, EMPCA, YYB, YYE,

*CRESC, DECRS, ATIVA, DIFTE, GLBAL, UM, DOIS, TRES, QUATR, CINCO,

*SEIS, SETE, OITO, NOVE, DEZ, ONZE, DOZE, TREZE, ZERO, TSTYY, TERM

DIMENSION YYDP(100), TITLO(10)

COMMON IMPR, LEIT, TRNSF, CHAVE, VALOR, CELLA, FREQC, TITLO,

*INFTO, AGORA, UM, DOIS, TRES, QUATR, CINCO, SEIS, SETE

*, OITO, NOVE, DEZ, ONZE, DOZE, TREZE, ZERO, NINTG, DECRS, INTGR,

*ENTRA, CRESC, ATIVA, INTNO, GLBAL, IGUAL, NNHUM, CONJT, SAIDA,

*PRMRO, NRMAL, DIFTE, ENTDD, ULTMO, ERLNG, TRSAO, PDSOR, TAMAN,

*UNFRM, PONTR, SUCSR, EMPCA, ELMTO, MAXMO, MINMO

COMMON IFLA, IFLB, YYB, YYE, NYY, TSTYY, JKYY, YYDP, NYYDP, YY

COMMON AGOYY, KKYY, JJYY, TERM, VETYY, CONTTY

DEFINE FILE 1(25, 201, U, KKYY)

DEFINE FILE 2(50, 250, U, JJYY)

INTEGER SERVEN, CAPADE

INTEGER EVENT1, EVENT2, EVENT3, STAESP, ESTAT1, ESTAT2, FILAAT, FILACA,

* CAMHAO, HORCHE, TEMPO, EQUIPE, FIM, AQUEC, FINAL, ESPERA

→ CALL INCLZ(1) ←

CALL DSTRB(INTCHE, ERLNG, 3, 1, 47981)

CALL DSTRB(IATEND, ERLNG, 10, 1, 58092)

CALL EVNTO(EVENT1, 0, ENTRA, 1, 0)

CALL EVNTO(EVENT2, 1, SAIDA, 1, 0)

CALL CRIEI(TRSAO, UM, CAMHAO)

HORCHE = UM

CALL SEJA(HORCHE, CAMHAO, -10)

CALL DFNAI(CONJT, 0, FILACA)

→ CALL DFNAI(CONJT, 0, ICAESP) ←

CALL SIGA(ICAESP, ISPERA, 1, 20, 1, 1)

CALL STATC(ESPERA, NINTG, 1, 12, 5, 5)

CALL CLQUE(CAMHAO, ULTMO, FILACA)

CALL CLQUE(CAMHAO, ULTMO, ICAESP)

CALL PRGME(EVENT1, DOIS)

CALL DFNAI(ENTDD, UM, SERVEN)

CAPADE = UM

→ CALL SEJA(CAPADE, SERVEN, CINCO) ←

CALL HSTRC(INOTA, 1, 2000)

50 CONTINUE

CALL ETAPA(IPROX)

GO TO (100, 200, 500), IPROX

CHEGADA DE CAMINHOS - EVENT1

```
C
100 CONTINUE
CALL CRIEI(TRSAO,UM,CAMHAO)
C
HORCHE = UM
CALL SEJA(HORCHE,CAMHAO,AGORA)
CALL CLQUE(CAMHAO,ULTMO,FILACA)
CALL CLQUE(CAMHAO,ULTMO,ICAESP)
TEMPO = AMSTA(INTCHE)
IF(TEMPO.GT.15) TEMPO = 15
CALL PRGME(EVENT1,TEMPO)
GO TO 50

C
C
C
FIM DE DESCARGA DOS CAMINHOES - EVENT2

C
200 CONTINUE
CAMHAO = TRNSF(1)
IF(.NOT.RMOVA(CAMHAO,ICAESP)) CALL ERRO(100)
CALL DSTRU(CAMHAO)
→ CALL SOME(UM,CAPADE,SERVEN) ←
IF(A(CAPADE,SERVEN).GT.CINCO) CALL ERRO(105)
GO TO 50

C
C
C
C
C
ATIVIDADES CONDICIONAIS

C
C
C
C
INICIO DE DESCARGA DOS CAMINHOES

C
500 CONTINUE
IF(A(CAPADE,SERVEN).EQ.ZERO.OR.O(TAMAN,FILACA).EQ.ZERO) GO TO 50
CALL SUBTR(UM,CAPADE,SERVEN)
IF(A(CAPADE,SERVEN).LT.ZERO) CALL ERRO(110)
CALL RTIRE(PRMRO,FILACA,CAMHAO)
IDIF = AGORA - A(HORCHE,CAMHAO)
CALL ANOTE(IDIF,1,INOTA)
CALL RGTRE(IDIF,ESPERA)
TEMPO = AMSTA(IATEND)
IF(TEMPO.LT.5) TEMPO = 5
IF(TEMPO.GT.45) TEMPO = 45
TRNSF(1) = CAMHAO
CALL PRGME(EVENT2,TEMPO)
GO TO 500

C
END
```

* XLSIM2 / V3 30MAR79 XM0560 YY30000 YYDP100 *

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO
DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR
A PRIMEIRA PALAVRA.

**-CORRER

-QUANTO TEMPO

**- 0

```

===== F I M D A S I M U L A C A O =====
===== F I M D A S I M U L A C A O =====
===== F I M D A S I M U L A C A O =====
===== F I M D A S I M U L A C A O =====
===== F I M D A S I M U L A C A O =====

```

```

===== AGORA = 0 ERRO NUMERO 1061 =====
===== AGORA = 0 ERRO NUMERO 1061 =====
===== AGORA = 0 ERRO NUMERO 1061 =====

```

Devido à estrutura de compilação de um programa em XLSIM2/FOR e como o erro ocorreu ainda na fase de inicialização (AGORA=0), não permitindo que se volte ao ponto inicial, o processo deve ser interrompido e uma nova sequência deverá ser tentada, quando devemos ligar as chaves para a depuração lógica que nos permitirá determinar onde ocorreu o erro, conforme é mostrado na página seguinte.

* XLSIM2 / V3 30MAR79 XM0560 YY30000 YYDP100 *

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-LIGAR

-QUE DESEJA LIGAR (CHAVES OU COMENTARIOS)

**-CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

**-8

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-CORRER

-QUANTO TEMPO

**- 0

DISTRIBUICAO	29686	-2	3	1	47981
DISTRIBUICAO	29694	-2	10	1	58092
EVENTO	29702	0	1	1	0
EVENTO	29709	1	-1	1	0
CRIEUM	-3	1	611		
CONTTY = 1	VETYY(SETE) =		611		
SEJA	1	611	-10		
DEFINAUM	-1	0	29716		

```

===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====

```

```

===== AGORA = 0 ERRO NUMERO 1061 =====
===== AGORA = 0 ERRO NUMERO 1061 =====
===== AGORA = 0 ERRO NUMERO 1061 =====

```

O exemplo apresentado é bastante simples e talvez não necessitasse de tal procedimento, já que o erro era na depuração do segundo conjunto, conforme mostra o erro e a listagem de depuração. No entanto, para o caso de um modelo complexo, cuja inicialização fosse mais extensa, o auxílio deste mecanismo torna-se mais necessário.

Eliminado o erro (uso indevido da constante CØNJT; foi digitado CØNJT, conforme pode ser observado na listagem do programa na página 63), o programa deve ser novamente compilado, linkeditado e executado.

Devemos abrir um parênteses e lembrar que, estando-se a trabalhar num sistema computacional conversacional (terminal remoto), estas etapas de alteração do programa, compilação, linkedição e execução, são realizadas de forma interativa, não havendo, portanto, quebra no processo global.

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-CORRER

-QUANTO TEMPO

**- 0

```

===== F I M D A S I M U L A C A O =====
===== F I M D A S I M U L A C A O =====
===== F I M D A S I M U L A C A O =====
===== F I M D A S I M U L A C A O =====
===== F I M D A S I M U L A C A O =====

```

```

===== AGORA = 0 ERRO NUMERO 1180 =====
===== AGORA = 0 ERRO NUMERO 1180 =====
===== AGORA = 0 ERRO NUMERO 1180 =====

```

Ainda assim novo erro ocorre. Repetimos as sequências anteriores, eliminamos o erro, executamos novamente, até que consigamos passar pela inicialização do modelo, sem erros, quando então seguimos utilizando o sistema. Desta feita, para se evitar a repetição da listagem de opções oferecidas pelo sistema, usamos o procedimento "DESLIGAR COMENTÁRIOS". Logo após, para o caso de ocorrer um erro antes do término da corrida, usamos o procedimento "TIRAR UMA FOTOGRAFIA DAS CONDIÇÕES ATUAIS DA CORRIDA". O erro de fato ocorre. O processo não é interrompido aqui. Usando-se a opção "RESTABELECE AS CONDIÇÕES INICIAIS DESTA CORRIDA", voltamos a simulação para o tempo 0. Mandamos o modelo correr novamente até um tempo anterior à ocorrência do erro, ligamos as chaves e podemos, então, observar onde o erro ocorreu. Tudo isto pode ser observado nas páginas seguintes.

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-LIGAR

-QUE DESEJA LIGAR (CHAVES OU COMENTARIOS)

**-CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

**-8

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-CORRER

-QUANTO TEMPO

**- 0

DISTRIBUICAO	29686	-2	3	1	47981		
DISTRIBUICAO	29694	-2	10	1	58092		
EVENO	29702	0	1	1	0		
EVENO	29709	1	-1	1	0		
CRUEUM	-3	1	611				
CONTTY = 1	VETYY(SETE) =		611				
S/EJA	1	611	-10				
DEFINAUM	-1	0	29716				
DEFINAUM	-1	0	29720				
SIGA	29720	29724	1	20	1	1	
ESTADISTICA	29759	2	1	12	5	5	
COLOQUE	611	-2	29716				
COLOQUE	611	-2	29720				
PROGRAME EVENTO =	29702	PARA AGORA +		2			
DEFINAUM	-2	1	29789				

===== F I M DA S I M U L A C A O =====
 ===== F I M DA S I M U L A C A O =====
 ===== F I M DA S I M U L A C A O =====
 ===== F I M DA S I M U L A C A O =====
 ===== F I M DA S I M U L A C A O =====

===== AGORA = 0 ERRO NUMERO 1180 =====
 ===== AGORA = 0 ERRO NUMERO 1180 =====
 ===== AGORA = 0 ERRO NUMERO 1180 =====

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-CORRER

-QUANTO TEMPO

**- 0

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS
- TIRAR UMA FOTOGRAFIA DAS CONDIC0ES ATUAIS DA CORRIDA
- LIMPEZA DE ESTATISTICAS E/OU HISTORICOS
- RELATORIO(IMPRESSAO) DE ESTATISTICAS E/OU GRAFICOS
- AUTOCORRELACAO(CORRELOGRAMAS E/OU CORRELOGRAMAS)
- RESTABELECEER AS CONDIC0ES INICIAIS DESTA CORRIDA
- TESTES DE KOLMOGOROV-SMIRNOV
- ALTERAR ATRIBUTO DE ENTIDADE
- MODIFICAR CONJUNTOS(RETIRAR OU COLOCAR ELEMENTOS
- REPROGRAMAR EVENTOS(PROGRAMAR NOVOS OU APAGAR EVENTOS JA PROGRAMADOS)
- ATIVAR CELULAS
- DESATIVAR CELULAS
- TROCAR PARAMETROS DE DISTRIBUICOES
- ANOTAR, EM HISTORICOS, PARAMETROS(MEDIA,MAXIMO,MINIMO, DESVIO PADRAO, TAMANHO) DAS ESTATISTICAS USADAS
- DEFINIR HISTORICOS
- MUDAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- REINICIALIZAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- GERAR VARIAVEIS ANTITETICAS
- CANCELAR A GERACAO DE VARIAVEIS ANTITETICAS
- PROCURAR ELEMENTOS EM CONJUNTOS OU GRUPOS
- LISTAR VALOR(ES) DE ATRIBUTO(S) DE ELEMENTO(S)
- PERMUTAR POSICAO DE ELEMENTO DENTRO DE CONJUNTO
- ORDENAR CONJUNTO(S)
- DESTRUIR ELEMENTO(S) CRIADO(S)
- FINALIZAR A ANALISE

--*--OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

--*--DESLIGAR

-QUE DESEJA DESLIGAR (CHAVES OU COMENTARIOS)

--*--COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--*--TIRAR

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--*--CORRER

-QUANTO TEMPO

--*-- 1440

```

===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====

```

```

===== AGORA = 7 ERRO NUMERO 1190 =====
===== AGORA = 7 ERRO NUMERO 1190 =====
===== AGORA = 7 ERRO NUMERO 1190 =====

```

```

=== IMPRESSAO DAS ESTATISTICAS POR ORDEM DE DEFINICAO DAS MESMAS ===
=== IMPRESSAO DAS ESTATISTICAS POR ORDEM DE DEFINICAO DAS MESMAS ===
=== IMPRESSAO DAS ESTATISTICAS POR ORDEM DE DEFINICAO DAS MESMAS ===

```

** IMPRESSAO DA ESTATISTICA *****
 29724

AMOSTRA TAMANHO = 7 MENOR = 1 MAIOR = 3
 MEDIA = 1.86 DESVIO PADRAO = 0.64

HISTOGRAMA

INTERVALO	FREQUENCIA
(, 1)	0
(1, 2)	29
(2, 3)	57
(3, 4)	14
(4, 5)	0

** IMPRESSAO DA ESTATISTICA *****
29759

AMOSTRA TAMANHO = 3 MENOR = 0 MAIOR = 10
MEDIA = 3.33 DESVIO PADRAO = 5.77

HISTOGRAMA

INTERVALO	FREQUENCIA	
(, 5)	67	I*****
(5, 10)	0	I
(10, 15)	33	I*****
(15, 20)	0	I
(20, 25)	0	I
(25, 30)	0	I
(30, 35)	0	I
(35, 40)	0	I
(40, 45)	0	I
(45, 50)	0	I
(50, 55)	0	I
(55,)	0	I

-ENTRE COM A OPCAO DESEJADA - AGORA = 7

--RESTABELECE

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--CORRER

-QUANTO TEMPO

-- 5

-ENTRE COM A OPCAO DESEJADA - AGORA = 5

--LIGAR

-QUE DESEJA LIGAR (CHAVES OU COMENTARIOS)

--CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

--8

-ENTRE COM A OPCAO DESEJADA - AGORA = 5

--CORRER

-QUANTO TEMPO

-- 10

```

ETAPA          1
AGORA =        6  EVENTO = 29702
CRIEUM         -3   1   631
CONTEY =      3  VETYY(SETE) = 631
SEJA          1   631   6
COLOQUE       631  -2  29716
COLOQUE       631  -2  29720
AMOSTRADE     29686  2
PROGRAME EVENTO = 29702  PARA AGORA + 2
ETAPA          3
AGORA =        6  ATIVIDADES CONDICIONAIS
A              1  29789   3
O             -3  29716   1
SUBTRAIA      1   1  29789
A              1  29789   2
RETIRE        -1  29716  631
A              1   631   6
ANOTE         0   1  29793
REGISTRE      0  29759
AMOSTRADE     29694  1
PROGRAME EVENTO = 29709  PARA AGORA + 5
TRANSFERENCIA 1 = 631
A              1  29789   2
O             -3  29716   0
ETAPA          2
AGORA =        7  EVENTO = 29709
TRANSFERENCIA 1 = 617
REMOVA        617   29720   T
DESTRUA       617
I = 1         VETYY(SETE) = 611
I = 2         VETYY(SETE) = 631

```

```

===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====
===== F I M DA S I M U L A C A O =====

```

```

===== AGORA = 7 ERRO NUMERO 1190 =====
===== AGORA = 7 ERRO NUMERO 1190 =====
===== AGORA = 7 ERRO NUMERO 1190 =====

```

```

=== IMPRESSAO DAS ESTATISTICAS POR ORDEM DE DEFINICAO DAS MESMAS ===
=== IMPRESSAO DAS ESTATISTICAS POR ORDEM DE DEFINICAO DAS MESMAS ===
=== IMPRESSAO DAS ESTATISTICAS POR ORDEM DE DEFINICAO DAS MESMAS ===

```

```

** IMPRESSAO DA ESTATISTICA *****
      29724

```

```

AMOSTRA TAMANHO =      7      MENOR =      1      MAIOR =      3
      MEDIA =      1.86      DESVIO PADRAO =      0.64

```

```

HISTOGRAMA
INTERVALO      FREQUENCIA
(      ,      1)      0      I
(      1,      2)      29      I*****
(      2,      3)      57      I*****
(      3,      4)      14      I*****
(      4,      5)      0      I
(      5,      6)      0      I
(      6,      7)      0      I
(      7,      8)      0      I
(      8,      9)      0      I
(      9,     10)      0      I
(     10,     11)      0      I
(     11,     12)      0      I
(     12,     13)      0      I
(     13,     14)      0      I
(     14,     15)      0      I

```

```

** IMPRESSAO DA ESTATISTICA *****
      29759

```

```

AMOSTRA TAMANHO =      3      MENOR =      0      MAIOR =     12
      MEDIA =      4.00      DESVIO PADRAO =      6.93

```

```

HISTOGRAMA
INTERVALO      FREQUENCIA
(      ,      5)      67      I*****
(      5,     10)      0      I
(     10,     15)      33      I*****
(     15,     20)      0      I
(     20,     25)      0      I
(     25,     30)      0      I
(     30,     35)      0      I
(     35,     40)      0      I
(     40,     45)      0      I
(     45,     50)      0      I
(     50,     55)      0      I
(     55,      )      0      I

```

```

-ENTRE COM A OPCAO DESEJADA - AGORA =      7

```

```

--FINALIZAR

```

Aqui novamente devemos interromper o uso do sistema, alterar o programa, compilá-lo, linkeditá-lo e executá-lo. A corrida consegue, finalmente, chegar ao final, sem erros. Podemos, então, utilizar o procedimento "RELATÓRIO (IMPRESSÃO) DE ESTATÍSTICAS E/OU GRÁFICOS", para observarmos os resultados. Caso eles sejam coerentes, podemos iniciar um processo de validação mais apurado, através de algum teste estatístico apropriado (isto será mostrado mais adiante). Caso os resultados não sejam bons, devemos então repetir algumas das sequências anteriores, até que consigamos resultados mais confiáveis. Esta última sequência é apresentada nas páginas seguintes.

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-CORRER

-QUANTO TEMPO

**- 0

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS
- TIRAR UMA FOTOGRAFIA DAS CONDICÕES ATUAIS DA CORRIDA
- LIMPEZA DE ESTATÍSTICAS E/OU HISTÓRICOS
- RELATÓRIO (IMPRESSÃO) DE ESTATÍSTICAS E/OU GRÁFICOS
- AUTOCORRELAÇÃO (CORRELOGRAMAS E/OU CORRELOBLOCOS)
- RESTABELECEER AS CONDIÇÕES INICIAIS DESTA CORRIDA
- TESTES DE KOLMOGOROV-SMIRNOV
- ALTERAR ATRIBUTO DE ENTIDADE
- MODIFICAR CONJUNTOS (RETIRAR OU COLOCAR ELEMENTOS)
- REPROGRAMAR EVENTOS (PROGRAMAR NOVOS OU APAGAR EVENTOS JÁ PROGRAMADOS)
- ATIVAR CELULAS
- DESATIVAR CELULAS
- TROCAR PARAMETROS DE DISTRIBUIÇÕES
- ANOTAR, EM HISTÓRICOS, PARAMETROS (MÉDIA, MÁXIMO, MÍNIMO, DESVIO PADRÃO, TAMANHO) DAS ESTATÍSTICAS USADAS
- DEFINIR HISTÓRICOS
- MUDAR A SEMENTE DA SEQUÊNCIA DE VALORES ALEATORIOS
- REINICIALIZAR A SEMENTE DA SEQUÊNCIA DE VALORES ALEATORIOS
- GERAR VARIÁVEIS ANTITÉTICAS
- CANCELAR A GERAÇÃO DE VARIÁVEIS ANTITÉTICAS
- PROCURAR ELEMENTOS EM CONJUNTOS OU GRUPOS
- LISTAR VALOR(ES) DE ATRIBUTO(S) DE ELEMENTO(S)
- PERMUTAR POSIÇÃO DE ELEMENTO DENTRO DE CONJUNTO
- ORDENAR CONJUNTO(S)
- DESTRUIR ELEMENTO(S) CRIADO(S)
- FINALIZAR A ANÁLISE

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

-*OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

-*DESLIGAR

-QUE DESEJA DESLIGAR (CHAVES OU COMENTARIOS)

-*COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

-*TIRAR

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

-*CORRER

-QUANTO TEMPO

*- 1440

-ENTRE COM A OPCAO DESEJADA - AGORA = 1440

*-RELATORIO

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

*- 1

-ENTRE COM O NOME DA ESTATISTICA

*-FILA DE CAMINHOES

** IMPRESSAO DA ESTATISTICA FILA DE CAMINHOES
29724

AMOSTRA TAMANHO = 1438 MENOR = 0 MAIOR = 14
MEDIA = 3.98 DESVIO PADRAO = 2.36

HISTOGRAMA

INTERVALO	FREQUENCIA	
(, 1)	4	I*****
(1, 2)	10	I*****
(2, 3)	14	I*****
(3, 4)	19	I*****
(4, 5)	17	I*****
(5, 6)	12	I*****
(6, 7)	9	I*****
(7, 8)	7	I*****
(8, 9)	3	I*****
(9, 10)	2	I****
(10, 11)	1	I**
(11, 12)	0	I
(12, 13)	0	I
(13, 14)	0	I

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

**- 2

-ENTRE COM O NOME DA ESTATISTICA

**-TEMPOS DE ESPERA DOS CAMINHOS

** IMPRESSAO DA ESTATISTICA TEMPOS DE ESPERA DOS CAMINHOS
29759

AMOSTRA TAMANHO = 452 MENOR = 0 MAIOR = 20
MEDIA = 1.73 DESVIO PADRAO = 3.60

HISTOGRAMA

INTERVALO	FREQUENCIA	
(, 5)	87	I*****
(5, 10)	8	I***
(10, 15)	2	I
(15, 20)	2	I
(20, 25)	1	I
(25, 30)	0	I
(30, 35)	0	I
(35, 40)	0	I
(40, 45)	0	I
(45, 50)	0	I
(50, 55)	0	I
(55,)	0	I

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

**- 0

-QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

**- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 1440

**-FINALIZAR

IV.3.2. DEFINIÇÃO DO PERÍODO TRANSIENTE

Quando estivermos interessados na definição do período transiente, após termos passado pelos passos do exemplo 1, poderemos utilizar a regra prática sugerida por Flávio Faria¹³, bastando para isto, antes de iniciarmos a corrida, ainda na fase de inicialização, definirmos um histórico, após o que poderemos correr o modelo e, a intervalos regulares, anotarmos o parâmetro desejado (no caso média) e limpamos a estatística. Isto será feito até conseguirmos um número razoável de pontos, quando então pediremos a impressão do gráfico, que nos permitirá a visualização da fase transiente. Estas etapas, com o uso dos adequados procedimentos, são apresentadas nas páginas seguintes. Tendo em vista que a listagem destes procedimentos seria bastante extensa e repetitiva, mudando-se apenas o relógio AGORA, resolvemos apresentar apenas parte dela, dando um salto do tempo AGORA=20 para o tempo AGORA=600.

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

--CORRER

-QUANTO TEMPO

-- 0

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS
- TIRAR UMA FOTOGRAFIA DAS CONDICAOES ATUAIS DA CORRIDA
- LIMPEZA DE ESTATISTICAS E/OU HISTORICOS
- RELATORIO(IMPRESSAO) DE ESTATISTICAS E/OU GRAFICOS
- AUTOCORRELACAO(CORRELOGRAMAS E/OU CORRELOBLOCOS)
- RESTABELECEER AS CONDICAOES INICIAIS DESTA CORRIDA
- TESTES DE KOLMOGOROV-SMIRNOV
- ALTERAR ATRIBUTO DE ENTIDADE
- MODIFICAR CONJUNTOS(RETIRAR OU COLOCAR ELEMENTOS)
- REPROGRAMAR EVENTOS(PROGRAMAR NOVOS OU APAGAR EVENTOS JA PROGRAMADOS)
- ATIVAR CELULAS
- DESATIVAR CELULAS
- TROCAR PARAMETROS DE DISTRIBUICOES
- ANOTAR, EM HISTORICOS, PARAMETROS(MEDIA, MAXIMO, MINIMO, DESVIO PADRAO, TAMANHO) DAS ESTATISTICAS USADAS
- DEFINIR HISTORICOS
- MUDAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- REINICIALIZAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- GERAR VARIAVEIS ANTITETICAS
- CANCELAR A GERACAO DE VARIAVEIS ANTITETICAS
- PROCURAR ELEMENTOS EM CONJUNTOS OU GRUPOS
- LISTAR VALOR(ES) DE ATRIBUTO(S) DE ELEMENTO(S)
- PERMUTAR POSICAO DE ELEMENTO DENTRO DE CONJUNTO
- ORDENAR CONJUNTO(S)
- DESTRUIR ELEMENTO(S) CRIADO(S)
- FINALIZAR A ANALISE

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

--DESLIGAR

-QUE DESEJA DESLIGAR (CHAVES OU COMENTARIOS)

--COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--DEFINIR

-QUANTAS SERIES DE VALORES TERA O HISTORICO

--1

-QUAL O NUMERO MAXIMO DE VALORES A SEREM ANOTADOS

-- 100

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--CORRER

-QUANTO TEMPO

-- 20

-ENTRE COM A OPCAO DESEJADA - AGORA = 20

--ANOTAR

-QUE PARAMETRO DESEJA ANOTAR (MEDIA, MAXIMO, MINIMO, DESVIO PADRAO, TAMANHO OU NENHUM)

--MEDIA

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA)

-- 2

-QUAL HISTORICO SERA USADO (DE A ORDEM DE DEFINICAO DO MESMO)

-- 2

-EM QUAL DAS SERIES SERA ANOTADO

--1

-QUE PARAMETRO DESEJA ANOTAR (MEDIA, MAXIMO, MINIMO, DESVIO PADRAO, TAMANHO OU NENHUM)

--NENHUM

-ENTRE COM A OPCAO DESEJADA - AGORA = 20

--LIMPEZA

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 2

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 0

-QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 20

--CORRER

-QUANTO TEMPO

-- 20

-ENTRE COM A OPCAO DESEJADA - AGORA = 600

--ANOTAR

-QUE PARAMETRO DESEJA ANOTAR (MEDIA, MAXIMO, MINIMO,
DESVIO PADRAO, TAMANHO OU NENHUM)

--MEDIA

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA)

-- 2

-QUAL HISTORICO SERA USADO (DE A ORDEM DE DEFINICAO DO MESMO)

-- 2

-EM QUAL DAS SERIES SERA ANOTADO

--1

-QUE PARAMETRO DESEJA ANOTAR (MEDIA, MAXIMO, MINIMO,
DESVIO PADRAO, TAMANHO OU NENHUM)

--NENHUM

-ENTRE COM A OPCAO DESEJADA - AGORA = 600

--LIMPEZA

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 2

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 0

-QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

--RELATORIO

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 0

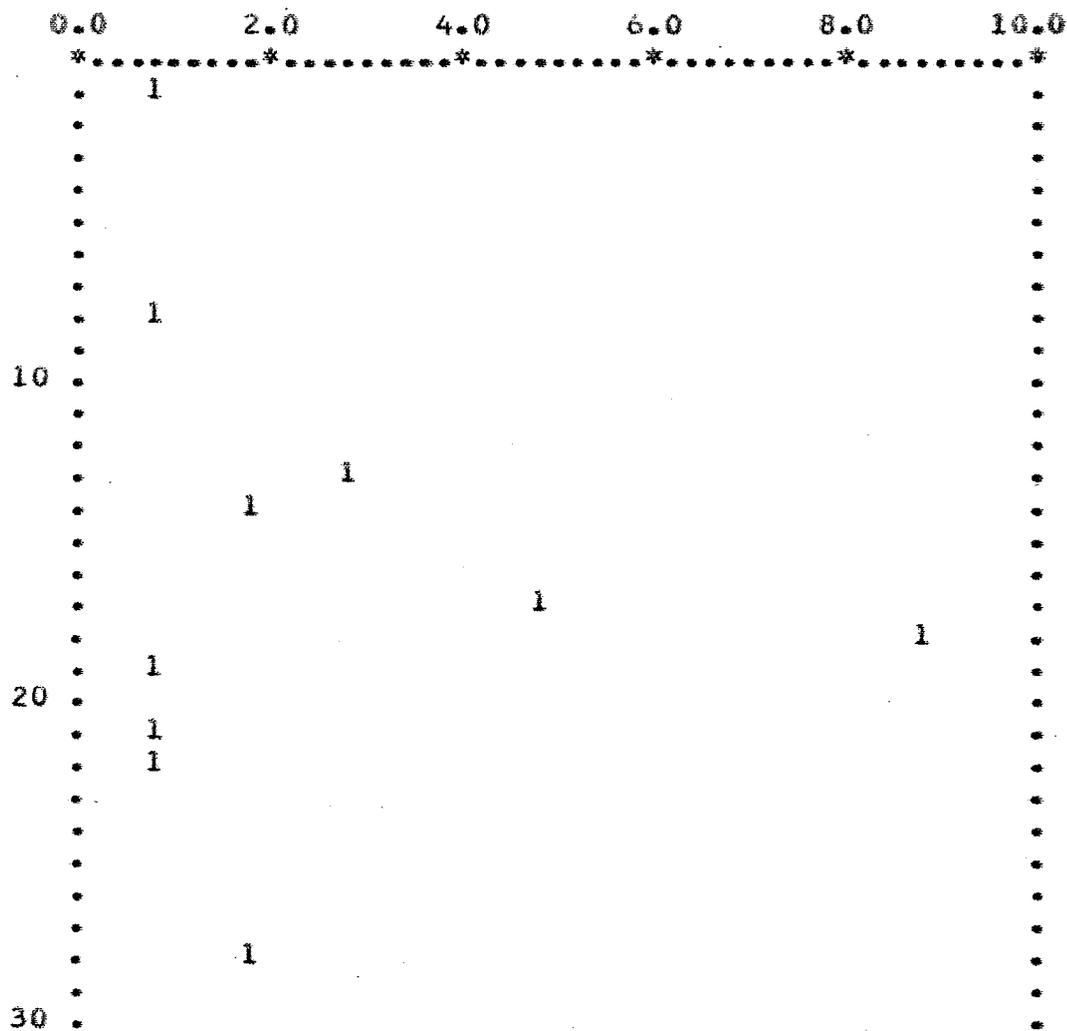
-QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 2

-VALORES AGRUPADOS EM NUMERO NAO SUPERIOR A

-- 1

GRAFICO COMPARATIVO DAS CORRIDAS ANOTADAS NO HISTORICO 29800
VALORES AGRUPADOS DE 1 EM 1



-QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 600

--FINALIZAR

IV.3.3. ANÁLISE DE AUTO-CORRELAÇÃO

Caso tenhamos que determinar o efeito da auto-correlação no modelo, podemos definir um histórico previamente, bem como o comando para anotação da variável de interesse. Observe aqui, que a definição do histórico é feita "a priori", bem como o comando para anotação, tendo em vista que a variável a ser analisada é aleatória e não saberíamos a hora da sua ocorrência. Além do mais, estamos interessados em todas as ocorrências desta variável, pois pretendemos justamente determinar o efeito que cada uma tem sobre as outras. Finalizada a corrida, podemos utilizar o procedimento "AUTOCORRELAÇÃO (CORRELOGRAMAS E/OU CORRELOBLOCOS)" e obter a definição do "lag" necessário à anotação da variável de interesse, com o consequente ganho de tempo nas corridas futuras. Esta opção é ilustrada a seguir:

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
 - LIGAR CHAVES OU COMENTARIOS
 - DESLIGAR CHAVES OU COMENTARIOS
- ENTRE COM A OPCAO DESEJADA - AGORA = 0
- **-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.
- **-CORRER
- QUANTO TEMPO
- **- 1440

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS
- TIRAR UMA FOTOGRAFIA DAS CONDIC0ES ATUAIS DA CORRIDA
- LIMPEZA DE ESTATISTICAS E/OU HISTORICOS
- RELATORIO(IMPRESSAO) DE ESTATISTICAS E/OU GRAFICOS
- AUTOCORRELACAO(CORRELOGRAMAS E/OU CORRELOBLOCOS)
- RESTABELECEER AS CONDIC0ES INICIAIS DESTA CORRIDA
- TESTES DE KOLMOGOROV-SMIRNOV
- ALTERAR ATRIBUTO DE ENTIDADE
- MODIFICAR CONJUNTOS(RETIRAR OU COLOCAR ELEMENTOS
- REPROGRAMAR EVENTOS(PROGRAMAR NOVOS OU APAGAR EVENTOS JA PROGRAMADOS)
- ATIVAR CELULAS
- DESATIVAR CELULAS
- TROCAR PARAMETROS DE DISTRIBUIC0ES
- ANOTAR, EM HISTORICOS, PARAMETROS(MEDIA,MAXIMO,MINIMO, DESVIO PADRAO, TAMANHO) DAS ESTATISTICAS USADAS
- DEFINIR HISTORICOS
- MUDAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- REINICIALIZAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- GERAR VARIAVEIS ANTITETICAS
- CANCELAR A GERACAO DE VARIAVEIS ANTITETICAS
- PROCURAR ELEMENTOS EM CONJUNTOS OU GRUPOS
- LISTAR VALOR(ES) DE ATRIBUTO(S) DE ELEMENTO(S)
- PERMUTAR POSICAO DE ELEMENTO DENTRO DE CONJUNTO
- ORDENAR CONJUNTO(S)
- DESTRUIR ELEMENTO(S) CRIADO(S)
- FINALIZAR A ANALISE

-ENTRE COM A OPCAO DESEJADA - AGORA = 1440

-*-AUTOCORRELACAO

-DESEJA CORRELOGRAMA (SIM OU NAO)

-*-SIM

-QUAL SERIE

-*-1

-QUAL HISTORICO SERA USADO (DE A ORDEM DE DEFINICAO DO MESMO)

-*- 1

-QUAL LAG MAXIMO

-*-20

ROTINA DE ANALISE DE AUTOCORRELACAO

HISTORICO 29793 - SERIE 1

AMOSTRA TAMANHO = 452 MENOR = 0 MAIOR = 20
 MEDIA = 1.73 DESVIO PADRAO = 3.60

VALOR DO FATOR DE CORRECAO DA VARIANCIA AMOSTRAL = 12.59353

VALORES DOS COEFICIENTES DE AUTOCORRELACAO DE LAG=1 ATE LAG= 20

LAG	VALOR	CORRELOGRAMA										
		-1.0	-0.8	-0.6	-0.4	-0.2	-0.0	0.2	0.4	0.6	0.8	1.0
		*.....*	*.....*	*.....*	*.....*	*.....*	*.....*	*.....*	*.....*	*.....*	*.....*
1	0.862	*****	*****	*****	*****	*****	.
2	0.773	*****	*****	*****	*****	*****	.
3	0.702	*****	*****	*****	*****	*****	.
4	0.633	*****	*****	*****	*****	*****	.
5	0.567	*****	*****	*****	*****	*****	.
6	0.505	*****	*****	*****	*****	*****	.
7	0.437	*****	*****	*****	*****	*****	.
8	0.375	*****	*****	*****	*****	*****	.
9	0.322	*****	*****	*****	*****	*****	.
10	0.264	*****	*****	*****	*****	*****	.
11	0.207	*****	*****	*****	*****	*****	.
12	0.165	*****	*****	*****	*****	*****	.
13	0.119	**	**	**	**	**	.
14	0.088	**	**	**	**	**	.
15	0.036
16	-.007
17	-.047	*	*	*	*	*	.
18	-.060	*	*	*	*	*	.
19	-.066	*	*	*	*	*	.
20	-.078	*	*	*	*	*	.

-DESEJA CORRELOGRAMA (SIM OU NAO)

--NAO

DESEJA CORRELOBLOCO (SIM OU NAO)

--SIM

-QUAL SERIE

--1

-QUAL HISTORICO SERA USADO (DE A ORDEM DE DEFINICAO DO MESMO)

-- 1

-QUAL O NUMERO MAXIMO DE OBSERVACOES POR BLOCO

-- 20

HISTORICO 29793 SERIE 1

VALORES DOS COEFICIENTES DE AUTOCORRELACAO DE LAG=1
PARA DIVERSOS TAMANHOS DE BLOCOS DE OBSERVACOES

TAMANHO DO BLOCO COEF. AUTOCORRELACAO

1	0.86237
2	0.84704
4	0.76736
8	0.58273
16	0.19389

GRAFICO DE COEF. AUTOCORRELACAO.VS.TAMANHO DE BLOCO

TAMANHO	VALORES DA AUTOCORRELACAO										
	-1.0	-0.8	-0.6	-0.4	-0.2	-0.0	0.2	0.4	0.6	0.8	1.0
*.....*.....*.....*.....*.....*.....*.....*										
1	.	*****									.
2	.	*****									.
4	.	*****									.
8	.	*****									.
16	.	*****									.

DESEJA CORRELOBLOCO (SIM OU NAO)

--NAO

-ENTRE COM A OPCAO DESEJADA - AGORA = 1440

--FINALIZAR

IV.3.4. TESTES ESTATÍSTICOS

Se, após a sequência do primeiro exemplo, tivermos uma estatística de dados históricos da variável de interesse, podemos testar a aderência dos resultados a esta distribuição histórica. Isto é possível, através do teste de Kolmogorov-Smirnov, apresentado na sequência seguinte:

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

--CORRER

-QUANTO TEMPO

-- 1440

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS
- TIRAR UMA FOTOGRAFIA DAS CONDIC0ES ATUAIS DA CORRIDA
- LIMPEZA DE ESTATISTICAS E/OU HISTORICOS
- RELATORIO(IMPRESSAO) DE ESTATISTICAS E/OU GRAFICOS
- AUTOCORRELACAO(CORRELOGRAMAS E/OU CORRELOBLOCOS)
- RESTABELECEER AS CONDIC0ES INICIAIS DESTA CORRIDA
- TESTES DE KOLMOGOROV-SMIRNOV
- ALTERAR ATRIBUTO DE ENTIDADE
- MODIFICAR CONJUNTOS(RETIRAR OU COLOCAR ELEMENTOS
- REPROGRAMAR EVENTOS(PROGRAMAR NOVOS OU APAGAR EVENTOS JA PROGRAMADOS)
- ATIVAR CELULAS
- DESATIVAR CELULAS
- TROCAR PARAMETROS DE DISTRIBUIC0ES
- ADOTAR, EM HISTORICOS, PARAMETROS(MEDIA, MAXIMO, MINIMO, DESVIO PADRAO, TAMANHO) DAS ESTATISTICAS USADAS
- DEFINIR HISTORICOS
- MUDAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- REINICIALIZAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- GERAR VARIAVEIS ANTITETICAS
- CANCELAR A GERACAO DE VARIAVEIS ANTITETICAS
- PROCURAR ELEMENTOS EM CONJUNTOS OU GRUPOS
- LISTAR VALOR(ES) DE ATRIBUTO(S) DE ELEMENTO(S)
- PERMUTAR POSICAO DE ELEMENTO DENTRO DE CONJUNTO
- ORDENAR CONJUNTO(S)
- DESTRUIR ELEMENTO(S) CRIADO(S)
- FINALIZAR A ANALISE

--OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

--DESLIGAR

-QUE DESEJA DESLIGAR (CHAVES OU COMENTARIOS)

--COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 1440

--RELATORIO

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 2

-ENTRE COM O NOME DA ESTATISTICA

--TEMPOS DE ESPERA DOS CAMINHOS

** IMPRESSAO DA ESTATISTICA TEMPOS DE ESPERA DOS CAMINHOS
29759

AMOSTRA TAMANHO = 452 MENOR = 0 MAIOR = 20
MEDIA = 1.73 DESVIO PADRAO = 3.60

HISTOGRAMA

INTERVALO	FREQUENCIA
(, 5)	87
(5, 10)	8
(10, 15)	2
(15, 20)	2
(20, 25)	1
(25, 30)	0
(30, 35)	0
(35, 40)	0
(40, 45)	0
(45, 50)	0
(50, 55)	0
(55,)	0

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 0

-QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 1440

--FINALIZAR

IV.3.5. EXPERIMENTAÇÃO

Ao final de uma corrida, algumas alterações são feitas no modelo antes que a próxima corrida seja executada, dependendo do tipo de experimentação que está sendo feito. Tais alterações podem ser, por exemplo, em:

1) - número de entidades (máquinas, serventes, etc.), trajetória das transações através do sistema, prioridade nas filas;

2) - padrão de chegadas, taxas de atendimento ou qualquer outra alteração de parâmetro de distribuição;

3) - modificação de eventos, programação de novos ou eliminação de eventos já programados para acontecer.

Todas estas mudanças e outras podem ser facilmente obtidas com o sistema. A seguir, apresentamos duas sequências onde algumas destas facilidades são executadas.

Na primeira sequência, após serem corridas 100 unidades de tempo, volta-se o modelo à condição inicial, quando as seguintes alterações são realizadas:

1) - o primeiro atributo da primeira entidade é mudado de cinco para quatro;

2) - a primeira transação é retirada dos conjuntos aos quais está alocada e destruída;

3) - o evento de número 2, anteriormente programado para ocorrer às 2 unidades de tempo é reprogramado para ocorrer às 5 unidades de tempo;

4) - a primeira distribuição tem sua média alterada de 3 para 4;

5) - o modelo é corrido novamente para 100 unidades de tempo.

Na segunda sequência, após o modelo correr 100 unidades de tempo, as seguintes alterações são feitas, pela ordem:

- 1) - ordenar o segundo conjunto;
- 2) - procurar, no segundo conjunto, um elemento cujo primeiro atributo é mínimo;
- 3) - retirar este elemento do conjunto;
- 4) - apagar o evento, no qual o elemento retirado está envolvido;
- 5) - destruir este elemento;
- 6) - procurar, no segundo conjunto, outro elemento, cujo primeiro atributo é mínimo;
- 7) - posicionar este elemento em ordem decrescente do primeiro atributo;
- 8) - listar o valor do primeiro atributo da primeira entidade criada;
- 9) - alterar o valor deste atributo;
- 10) - correr mais 100 unidades de tempo;
- 11) - imprimir as estatísticas das medidas de desempenho do sistema.

Em ambas as sequências, as chaves são ligadas, em determinados instantes, para que se possa melhor observar as alterações efetuadas.

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-CORR ER

-QUANTO TEMPO

**- 0

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS
- TIRAR UMA FOTOGRAFIA DAS CONDICÕES ATUAIS DA CORRIDA
- LIMPEZA DE ESTATÍSTICAS E/OU HISTÓRICOS
- RELATÓRIO (IMPRESSÃO) DE ESTATÍSTICAS E/OU GRÁFICOS
- AUTOCORRELAÇÃO (CORRELOGRAMAS E/OU CORRELOBLOCOS)
- RESTABELECE AS CONDIÇÕES INICIAIS DESTA CORRIDA
- TESTES DE KOLMOGOROV-SMIRNOV
- ALTERAR ATRIBUTO DE ENTIDADE
- MODIFICAR CONJUNTOS (RETIRAR OU COLOCAR ELEMENTOS)
- REPROGRAMAR EVENTOS (PROGRAMAR NOVOS OU APAGAR EVENTOS JÁ PROGRAMADOS)
- ATIVAR CELULAS
- DESATIVAR CELULAS
- TROCAR PARÂMETROS DE DISTRIBUIÇÕES
- ANOTAR, EM HISTÓRICOS, PARÂMETROS (MÉDIA, MÁXIMO, MÍNIMO, DESVIO PADRÃO, TAMANHO) DAS ESTATÍSTICAS USADAS
- DEFINIR HISTÓRICOS
- MUDAR A SEMENTE DA SEQUÊNCIA DE VALORES ALEATORIOS
- REINICIALIZAR A SEMENTE DA SEQUÊNCIA DE VALORES ALEATORIOS
- GERAR VARIÁVEIS ANTITÉTICAS
- CANCELAR A GERAÇÃO DE VARIÁVEIS ANTITÉTICAS
- ~~PROCURAR ELEMENTOS EM CONJUNTOS OU GRUPOS~~
- LISTAR VALOR(ES) DE ATRIBUTO(S) DE ELEMENTO(S)
- PERMUTAR POSIÇÃO DE ELEMENTO DENTRO DE CONJUNTO
- ORDENAR CONJUNTO(S)
- DESTRUIR ELEMENTO(S) CRIADO(S)
- FINALIZAR A ANÁLISE

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

97

--OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO
DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR
A PRIMEIRA PALAVRA.

--DESLIGAR

-QUE DESEJA DESLIGAR (CHAVES OU COMENTARIOS)

--COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--LIGAR

-QUE DESEJA LIGAR (CHAVES OU COMENTARIOS)

--CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

--8

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--TIRAR

COMEC ECORRIDA

AGORA FOI ZERADA E OS TEMPOS DE OCORRENCIA DOS EVENTOS
PROGRAMADOS FORAM ALTERADOS DE ACORDO

TODAS AS ESTATISTICAS FORAM LIMPAS

TODOS OS HISTORICOS FORAM LIMPOS

TODAS AS SEQUENCIAS DE NUMEROS ALEATORIOS FORAM REINICIALIZADAS

O ESTADO ATUAL DO SISTEMA FOI GUARDADO PARA USO COMO

CONDICOES INICIAIS PARA AS PROXIMAS CORRIDAS

AGORA = 0

I = 1 VETYY = 611

I = 2 VETYY = 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--CORRER

-QUANTO TEMPO

-- 10

ETAPA		3			
AGORA =	0	ATIVIDADES	CONDICIONAIS		
A		1	29789	5	
O		-3	29716	1	
SUBTRAI A		1	1 29789		
A		1	29789	4	
RETIRE		-1	29716	611	
A		1	611	-10	
ANOTE		10	1 29793		
REGISTRE		10	29759		
AMOSTRADE		29694	11		
PROGRAME EVENTO =	29709	PARA AGORA +		11	
TRANSFERENCIA 1 =		611			
A		1	29789	4	
O		-3	29716	0	
ETAPA		1			
AGORA =	2	EVENTO =	29702		
CRIEUM		-3	1 621		
CONTY Y =	2	VETYY(SETE) =	621		
SEJA		1	621	2	
COLOQUE		621	-2 29716		
COLOQUE		621	-2 29720		
AMOSTRADE		29686	4		
PROGRAME EVENTO =	29702	PARA AGORA +		4	
ETAPA		3			
AGORA =	2	ATIVIDADES	CONDICIONAIS		
A		1	29789	4	
O		-3	29716	1	
SUBTRAI A		1	1 29789		
A		1	29789	3	
RETIRE		-1	29716	621	
A		1	621	2	
ANOTE		0	1 29793		
REGISTRE		0	29759		
AMOSTRADE		29694	2		
PROGRAME EVENTO =	29709	PARA AGORA +		5	
TRANSFERENCIA 1 =		621			
A		1	29789	3	
O		-3	29716	0	
ETAPA		1			
AGORA =	6	EVENTO =	29702		
CRIEUM		-3	1 631		
CONTY Y =	3	VETYY(SETE) =	631		
SEJA		1	631	6	
COLOQUE		631	-2 29716		
COLOQUE		631	-2 29720		
AMOSTRADE		29686	2		
PROGRAME EVENTO =	29702	PARA AGORA +		2	
ETAPA		3			
AGORA =	6	ATIVIDADES	CONDICIONAIS		
A		1	29789	3	
O		-3	29716	1	
SUBTRAI A		1	1 29789		
A		1	29789	2	
RETIRE		-1	29716	631	
A		1	631	6	
ANOTE		0	1 29793		
REGISTRE		0	29759		
AMOSTRADE		29694	1		
PROGRAME EVENTO =	29709	PARA AGORA +		5	
TRANSFERENCIA 1 =		631			
A		1	29789	2	
O		-3	29716	0	

```

ETAPA                2
AGORA = 7  EVENTO = 29709
  TRANSFERENCIA 1 = 621
  REMOVA        621      29720  T
  DESTRUA      621
I = 1  VETYY(SETE) = 611
I = 2  VETYY(SETE) = 631
ADICIONE      1      1 29789
A            1 29789      3
ETAPA        3
AGORA = 7  ATIVIDADES CONDICIONAIS
A            1 29789      3
O           -3 29716      0
ETAPA        1
AGORA = 8  EVENTO = 29702
CRIEUM      -3      1 641
CONTTY = 3  VETYY(SETE) = 641
SEJA        1 641      8
COLOQUE     641     -2 29716
COLOQUE     641     -2 29720
AMOSTRADE   29686      0
PROGRAME EVENTO = 29702  PARA AGORA + 0
ETAPA        1
AGORA = 8  EVENTO = 29702
CRIEUM      -3      1 647
CONTTY = 4  VETYY(SETE) = 647
SEJA        1 647      8
COLOQUE     647     -2 29716
COLOQUE     647     -2 29720
AMOSTRADE   29686      1
PROGRAME EVENTO = 29702  PARA AGORA + 1
ETAPA        3
AGORA = 8  ATIVIDADES CONDICIONAIS
A            1 29789      3
O           -3 29716      2
SUBTRAI A    1      1 29789
A            1 29789      2
RETIRE      -1 29716     641
A            1 641      8
ANOTE        0      1 29793
REGISTRE     0 29759
AMOSTRADE   29694      7
PROGRAME EVENTO = 29709  PARA AGORA + 7
  TRANSFERENCIA 1 = 641
A            1 29789      2
O           -3 29716      1
SUBTRAI A    1      1 29789
A            1 29789      1
RETIRE      -1 29716     647
A            1 647      8
ANOTE        0      1 29793
REGISTRE     0 29759
AMOSTRADE   29694      2
PROGRAME EVENTO = 29709  PARA AGORA + 5
  TRANSFERENCIA 1 = 647
A            1 29789      1
O           -3 29716      0
ETAPA        1
AGORA = 9  EVENTO = 29702
CRIEUM      -3      1 661
CONTTY = 5  VETYY(SETE) = 661
SEJA        1 661      9
COLOQUE     661     -2 29716
COLOQUE     661     -2 29720
AMOSTRADE   29686      3
PROGRAME EVENTO = 29702  PARA AGORA + 3

```

ETAPA		3		
AGORA =	9	ATIVIDADES	CONDICIONAIS	
A		1	29789	1
D		-3	29716	1
SUBTRAI A		1	1 29789	
A		1	29789	0
RETIRE		-1	29716	661
A		1	661	9
ANOTE		0	1 29793	
REGISTRE		0	29759	
AMOSTRADE		29694	29	
PROGRAME EVENTO =	29709	PARA AGORA +		29
TRANSFERENCIA 1 =		661		
A		1	29789	0
D		-3	29716	0

-ENTRE COM A OPCAO DESEJADA - AGORA = 10

--*--DESLIGAR

-QUE DESEJA DESLIGAR (CHAVES OU COMENTARIOS)

--*--CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

--*--8

-ENTRE COM A OPCAO DESEJADA - AGORA = 10

--*--CORRER

-QUANTO TEMPO

--*-- 90

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--RELATORIO

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 1

-ENTRE COM O NOME DA ESTATISTICA

--FILA DE CAMINHOES

** IMPRESSAO DA ESTATISTICA FILA DE CAMINHOES
29724

AMOSTRA TAMANHO = 99 MENOR = 1 MAIOR = 7
MEDIA = 3.73 DESVIO PADRAO = 1.54

HISTOGRAMA

INTERVALO	FREQUENCIA
(0, 1)	0
(1, 2)	10
(2, 3)	13
(3, 4)	19
(4, 5)	21
(5, 6)	28
(6, 7)	4
(7, 8)	4
(8, 9)	0
(9, 10)	0
(10, 11)	0
(11, 12)	0
(12, 13)	0
(13, 14)	0
(14, 15)	0
(15, 16)	0
(16, 17)	0
(17, 18)	0
(18, 19)	0
(19,)	0

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

--*-- 2

-ENTRE COM O NOME DA ESTATISTICA

--*--TEMPOS DE ESPERA DOS CAMINHOS

** IMPRESSAO DA ESTATISTICA TEMPOS DE ESPERA DOS CAMINHOS
29759

AMOSTRA TAMANHO = 32 MENOR = 0 MAIOR = 10
MEDIA = 0.69 DESVIO PADRAO = 1.87

HISTOGRAMA

INTERVALO	FREQUENCIA	
(0, 5)	97	I*****
(5, 10)	0	I
(10, 15)	3	I*
(15, 20)	0	I
(20, 25)	0	I
(25, 30)	0	I
(30, 35)	0	I
(35, 40)	0	I
(40, 45)	0	I
(45, 50)	0	I
(50, 55)	0	I
(55,)	0	I

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

--*-- 0

-QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

--*-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--*--LIGAR

-QUE DESEJA LIGAR (CHAVES OU COMENTARIOS)

--*--CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

--*--8

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--RESTABELECE

RECOMECECORRIDA

TODAS AS SEQUENCIAS DE NUMEROS ALEATORIOS FORAM REINICIALIZADAS
CONDICOES INICIAIS FORAM RESTABELECIDAS

I = 1 VAUYY = 611
I = 2 VAUYY = 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--LIGAR

-QUE DESEJA LIGAR (CHAVES OU COMENTARIOS)

--CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

--8

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--LISTAR

-QUE TIPO DE ELEMENTO (ENTIDADE OU TRANSACAO)

--ENTIDADE

-QUAL A ORDEM DE DEFINICAO DESTE ELEMENTO

-- 1

-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 1

0 1 29789 5
-0 1 ATRIBUTO DA 1 ENTIDADE CRIADA OU DEFINIDA E = 5

-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--ALTERAR

-QUE ELEMENTO DESEJA ALTERAR (ENTIDADE, TRANSACAO OU NENHUM)

--ENTIDADE

-QUAL ENTIDADE (DE A ORDEM DE CRIACAO DA MESMA - 0 PARA PARAR

-- 1

-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 1

-QUAL O NOVO VALOR DO ATRIBUTO

-- 4

SEJA 1 29789 4
-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-QUAL ENTIDADE (DE A ORDEM DE CRIACAO DA MESMA - 0 PARA PARAR

-- 0

-QUE ELEMENTO DESEJA ALTERAR (ENTIDADE, TRANSACAO OU NENHUM)

--NENHUM

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--MODIFICAR

-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 1

-QUAL MODIFICACAO (RETIRAR OU COLOCAR)

--RETIRAR

-QUAL ELEMENTO (PRIMEIRO,ULTIMO,QUALQUER,DETERMINADO(S) OU TODOS)

--ULTIMO

RETIRE -2 29716 611
-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 2

-QUAL MODIFICACAO (RETIRAR OU COLOCAR)

--RETIRAR

-QUAL ELEMENTO (PRIMEIRO,ULTIMO,QUALQUER,DETERMINADO(S) OU TODOS)

--ULTIMO

RETIRE -2 29720 611
-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--DESTRUIR

-QUAL TRANSACAO (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 1

DESTRUA 611

-QUAL TRANSACAO (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--REPROGRAMAR

-QUE REPROGRAMACAO DESEJA (PROGRAMAR OU APAGAR EVENTOS OU NENHUMA DELAS)

--APAGAR

-QUAL EVENTO (DE A ORDEM DE DEFINICAO DO MESMO)

-- 1

-ALGUM ELEMENTO (ENTIDADE OU TRANSACAO) TRANSFERIDO NESTE EVENTO
 -(SIM OU NAO)

--NAO

LEIA 1
 29702 2
 T

APAGUE 1 29702 2
 -QUE REPROGRAMACAO DESEJA (PROGRAMAR OU APAGAR EVENTOS OU NENHUMA DELAS)

--PROGRAMAR

-QUAL EVENTO (DE A ORDEM DE DEFINICAO DO MESMO)

-- 1

-QUAL O TEMPO

-- 5

-ALGUM ELEMENTO (ENTIDADE OU TRANSACAO) TRANSFERIDO NESTE EVENTO
 -(SIM, NAO OU FIM)

--NAO

PROGRAMA EVENTO = 29702 PARA AGORA + 5
 -QUE REPROGRAMACAO DESEJA (PROGRAMAR OU APAGAR EVENTOS OU NENHUMA DELAS)

--NENHUMA

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--*-TROC AR

-QUAL DISTRIBUICAO (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

--*- 1

-QUAL O TIPO DA DISTRIBUICAO (NORMAL, ERLANG, UNIFORME OU EMPIRICA)

--*-ERLANG

-QUAL A MEDIA DE CADA ESTAGIO DA ERLANG

--*- 4

-QUAL O NUMERO DE ESTAGIOS DA ERLANG

--*- 1

DISTRIBUICAO 29686 -2 4 1 0
 -QUAL DISTRIBUICAO (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

--*- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

--*-CORRER

-QUANTO TEMPO

--*- 10

ETAPA		1		
AGORA =	5	EVENTO =	29702	
CRUEM		-3	1	619
CONTY Y =	1	VETYY(SETE) =	619	
SEJA		1	619	5
COLOQUE		619	-2	29716
COLOQUE		619	-2	29720
AMOSTRADE		29686	5	
PROGRAME EVENTO =	29702	PARA AGORA +		5
ETAPA		3		
AGORA =	5	ATIVIDADES CONDICIONAIS		
A		1	29789	4
O		-3	29716	1
SUBTRAIA		1	1	29789
A		1	29789	3
RETIRE		-1	29716	619
A		1	619	5
ANOTE		0	1	29793
REGISTRE		0	29759	
AMOSTRADE		29694	11	
PROGRAME EVENTO =	29709	PARA AGORA +		11
TRANSFERENCIA	1	=	619	
A		1	29789	3
O		-3	29716	0

-ENTRE COM A OPCAO DESEJADA - AGORA = 10

--DESLIGAR

-QUE DESEJA DESLIGAR (CHAVES OU COMENTARIOS)

--CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

--8

-ENTRE COM A OPCAO DESEJADA - AGORA = 10

--CORRER

-QUANTO TEMPO

-- 90

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--RELATORIO

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 1

-ENTRE COM O NOME DA ESTATISTICA

--FILA DE CAMINHOS

** IMPRESSAO DA ESTATISTICA FILA DE CAMINHOS

29724

AMOSTRA	TAMANHO =	92	MENOR =	0	MAIOR =	7
	MEDIA =	3.32	DESVIO PADRAO =	1.49		

HISTOGRAMA

INTERVALO	FREQUENCIA	
(0 1)	5	I*****
(1 2)	7	I*****
(2 3)	14	I*****
(3 4)	24	I*****
(4 5)	33	I*****
(5 6)	11	I*****
(6 7)	5	I*****
(7 8)	1	I*
(8 9)	0	I
(9 10)	0	I

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

--*-- 2

-ENTRE COM O NOME DA ESTATISTICA

--*--TEMPOS DE ESPERA DOS CAMINHOS

** IMPRESSAO DA ESTATISTICA TEMPOS DE ESPERA DOS CAMINHOS
29759

AMOSTRA TAMANHO = 24 MENOR = 0 MAIOR = 5
MEDIA = 0.96 DESVIO PADRAO = 1.43

HISTOGRAMA

INTERVALO	FREQUENCIA	
(5, 10)	96	I*****
(10, 15)	4	I*
(15, 20)	0	I
(20, 25)	0	I
(25, 30)	0	I
(30, 35)	0	I
(35, 40)	0	I
(40, 45)	0	I
(45, 50)	0	I
(50, 55)	0	I
(55,)	0	I

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

--*-- 0

-QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

--*-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--*--FINALIZAR

OPCOES OFERECIDAS NESTE MOMENTO

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 0

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-CORRER

-QUANTO TEMPO

**- 90

S I S T E M A A N A L I S A D O R

OPCOES OFERECIDAS

- CORRER O MODELO
- LIGAR CHAVES OU COMENTARIOS
- DESLIGAR CHAVES OU COMENTARIOS
- TIRAR UMA FOTOGRAFIA DAS CONDICoes ATUAIS DA CORRIDA
- LIMPEZA DE ESTATISTICAS E/OU HISTORICOS
- RELATORIO (IMPRESSAO) DE ESTATISTICAS E/OU GRAFICOS
- AUTOCORRELACAO (CORRELOGRAMAS E/OU CORRELOBLOCOS)
- RESTABELECEER AS CONDICoes INICIAIS DESTA CORRIDA
- TESTES DE KOLMOGOROV-SMIRNOV
- ALTERAR ATRIBUTO DE ENTIDADE
- MODIFICAR CONJUNTOS (RETIRAR OU COLOCAR ELEMENTOS)
- REPROGRAMAR EVENTOS (PROGRAMAR NOVOS OU APAGAR EVENTOS JA PROGRAMADOS)
- ATIVAR CELULAS
- DESBTIVAR CELULAS
- TROCAR PARAMETROS DE DISTRIBUICOES
- ANOTAR, EM HISTORICOS, PARAMETROS (MEDIA, MAXIMO, MINIMO, DESVIO PADRAO, TAMANHO) DAS ESTATISTICAS USADAS
- DEFINIR HISTORICOS
- MUDAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- REINICIALIZAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS
- GERAR VARIAVEIS ANTITETICAS
- CANCELAR A GERACAO DE VARIAVEIS ANTITETICAS
- PROCURAR ELEMENTOS EM CONJUNTOS OU GRUPOS
- LISTAR VALOR(ES) DE ATRIBUTO(S) DE ELEMENTO(S)
- PERMUTAR POSICAO DE ELEMENTO DENTRO DE CONJUNTO
- ORDENAR CONJUNTO(S)
- DESTRUIR ELEMENTO(S) CRIADO(S)
- FINALIZAR A ANALISE

-ENTRE COM A OPCAO DESEJADA - AGORA = 90

110

**-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR A PRIMEIRA PALAVRA.

**-DESLIGAR

-QUE DESEJA DESLIGAR (CHAVES OU COMENTARIOS)

**-COMENTARIOS

-ENTRE COM A OPCAO DESEJADA - AGORA = 90

**-LIGAR

-QUE DESEJA LIGAR (CHAVES OU COMENTARIOS)

**-CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

**-8

-ENTRE COM A OPCAO DESEJADA - AGORA = 90

**-CORRER

-QUANTO TEMPO

**- 10

```
ETAPA                2
AGORA = 94  EVENTO   = 29709
  TRANSFERENCIA 1 = 901
  REMOVA        901      29720      T
  DESTRUA      901
  I = 1        VETY(Y(SETE)) = 861
  I = 2        VETY(Y(SETE)) = 871
  ADICIONE          1      1 29789
  A                1 29789      3
  ETAPA            3
AGORA = 94  ATIVIDADES CONDICIONAIS
  A                1 29789      3
  O               -3 29716      0
  ETAPA            2
AGORA = 95  EVENTO   = 29709
  TRANSFERENCIA 1 = 871
  REMOVA        871      29720      T
  DESTRUA      871
  I = 1        VETY(Y(SETE)) = 861
  ADICIONE          1      1 29789
  A                1 29789      4
  ETAPA            3
AGORA = 95  ATIVIDADES CONDICIONAIS
  A                1 29789      4
  O               -3 29716      0
```

```

ETAPA = 1
AGORA = 99 EVENTO = 29702
CRIEUM -3 1 615
CONTY = 2 VETYY(SETE) = 615
SEJA 1 615 99
COLOQUE 615 -2 29716
COLOQUE 615 -2 29720
AMOSTRADE 29686 0
PROGRAMA EVENTO = 29702 PARA AGORA + 0
ETAPA = 1
AGORA = 99 EVENTO = 29702
CRIEUM -3 1 629
CONTY = 3 VETYY(SETE) = 629
SEJA 1 629 99
COLOQUE 629 -2 29716
COLOQUE 629 -2 29720
AMOSTRADE 29686 1
PROGRAMA EVENTO = 29702 PARA AGORA + 1
ETAPA = 3
AGORA = 99 ATIVIDADES CONDICIONAIS
A 1 29789 4
O -3 29716 2
SUBTRAI A 1 1 29789
A 1 29789 3
RETIRE -1 29716 615
A 1 615 99
ANOTE 0 1 29793
REGISTRE 0 29759
AMOSTRADE 29694 26
PROGRAMA EVENTO = 29709 PARA AGORA + 26
TRANSFERENCIA 1 = 615
A 1 29789 3
O -3 29716 1
SUBTRAI A 1 1 29789
A 1 29789 2
RETIRE -1 29716 629
A 1 629 99
ANOTE 0 1 29793
REGISTRE 0 29759
AMOSTRADE 29694 7
PROGRAMA EVENTO = 29709 PARA AGORA + 7
TRANSFERENCIA 1 = 629
A 1 29789 2
O -3 29716 0

```

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--ORDENAR

-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 2

-EM QUE ORDEM (CRESCENTE OU DECRESCENTE)

--DECRESCENTE

-QUAL ATRIBUTO (TAMANHO PODE SER USADO. NESTE CASO DIGITE -3

-- 1

ORDENE 29720 2 1
 -QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--*--PROCURAR

-O ELEMENTO PROCURADO PERTENCE A UM GRUPO OU CONJUNTO

--*--CONJUNTO

-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

--*-- 2

-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO)

--*-- 1

-QUAL O VALOR A SER COMPARADO (MAXIMO, MINIMO OU OUTRO QUALQUER)

--*--MINIMO

PROCURE 29720 1***** 861

-O ELEMENTO FOI ENCONTRADO. SUA ORDEM DE DEFINICAO E 1

-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

--*-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--*--MODIFICAR

-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

--*-- 2

-QUAL MODIFICACAO (RETIRAR OU COLOCAR)

--*--RETIRAR

-QUAL ELEMENTO (PRIMEIRO, ULTIMO, QUALQUER, DETERMINADO(S) OU TODOS)

--*--QUALQUER

-QUE TIPO DE ELEMENTO (ENTIDADE OU TRANSACAO)

--*--TRANSACAO

-QUAL ELEMENTO (DE A ORDEM DE DEFINICAO DO MESMO)

--*-- 1

REMOVA 861 29720 T

-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

--*-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--REPROGRAMAR

-QUE REPROGRAMACAO DESEJA (PROGRAMAR OU APAGAR EVENTOS OU NENHUMA DELAS)

--APAGAR

-QUAL EVENTO (DE A ORDEM DE DEFINICAO DO MESMO)

-- 2

-ALGUM ELEMENTO (ENTIDADE OU TRANSACAO) TRANSFERIDO NESTE EVENTO
 -(SIM OU NAO)

--SIM

-QUE TIPO DE ELEMENTO (ENTIDADE OU TRANSACAO)

--TRANSACAO

-QUAL ELEMENTO (DE A ORDEM DE DEFINICAO DO MESMO)

-- 1

-QUAL FOI A ORDEM DE TRANSFERENCIA DESTE ELEMENTO

-- 1

LEIA 1
 29702 100
 T

LEIA 2
 29709 105
 861
 T

APAGUE 2 29709 105
 861

-QUE REPROGRAMACAO DESEJA (PROGRAMAR OU APAGAR EVENTOS OU NENHUMA DELAS)

--NENHUMA

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--DESTRUIR

-QUAL TRANSACAO (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 1

DESTRUA 861
 I = 1 VETYY(SETE) = 615
 I = 2 VETYY(SETE) = 629

-QUAL TRANSACAO (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--PROCURAR

-O ELEMENTO PROCURADO PERTENCE A UM GRUPO OU CONJUNTO

--CONJUNTO

-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 2

-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO)

-- 1

-QUAL O VALOR A SER COMPARADO (MAXIMO, MINIMO OU OUTRO QUALQUER)

--MINIMO

PROCURE 29720 1***** 615
-O ELEMENTO FOI ENCONTRADO. SUA ORDEM DE DEFINICAO E 1

-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--PERMUTAR

-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 2

-QUAL O TIPO DE ELEMENTO A SER PERMUTADO

-(ENTIDADE, TRANSACAO, CONJUNTO, PONTEIRO OU GRUPO)

--TRANSACAO

-QUAL A ORDEM DE DEFINICAO DESTA ELEMENTO

-- 2

-EM QUE ORDEM (CRESCENTE OU DECRESCENTE)

--DECRESCENTE

-QUAL ATRIBUTO (TAMANHO PODE SER USADO CASO O ELEMENTO A SER PERMUTADO SEJA CONJUNTO OU GRUPO. NESTE CASO DIGITE -3)

-- 1

POSICIONE 629 29720 2 1
-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--LISTAR

-QUE TIPO DE ELEMENTO (ENTIDADE OU TRANSACAO)

--ENTIDADE

-QUAL A ORDEM DE DEFINICAO DESTE ELEMENTO

-- 1

-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 1

0 1 ATRIBUTO DA 1 ENTIDADE 1 29789 2 CRIADA OU DEFINIDA E = 2

-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 100

--ALTERAR

-QUE ELEMENTO DESEJA ALTERAR (ENTIDADE, TRANSACAO OU NENHUM)

--ENTIDADE

-QUAL ENTIDADE (DE A ORDEM DE CRIACAO DA MESMA - 0 PARA PARAR

-- 1

-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 1

-QUAL O NOVO VALOR DO ATRIBUTO

-- 3

SEJA 1 29789 3
-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)

-- 0

-QUAL ENTIDADE (DE A ORDEM DE CRIACAO DA MESMA - 0 PARA PARAR

-- 0

-QUE ELEMENTO DESEJA ALTERAR (ENTIDADE, TRANSACAO OU NENHUM)

--NENHUM

--CORR ER

-QUANTO TEMPO

-- 10

ETAPA 1
 AGORA = 100 EVENTO = 29702
 CRIEUM -3 1 619
 CONTTY = 3 VETYY(SETE) = 619
 SEJA 1 619 100
 COLOQUE 619 -2 29716
 COLOQUE 619 -2 29720
 AMOSTRADE 29686 5
 PROGRAME EVENTO = 29702 PARA AGORA + 5
 ETAPA 3

AGORA = 100 ATIVIDADES CONDICIONAIS
 A 1 29789 3
 O -3 29716 1
 SUBTRAIA 1 1 29789
 A 1 29789 2
 RETIRE -1 29716 619
 A 1 619 100
 ANOTE 0 1 29793
 REGISTRE 0 29759
 AMOSTRADE 29694 0
 PROGRAME EVENTO = 29709 PARA AGORA + 5
 TRANSFERENCIA 1 = 619
 A 1 29789 2
 O -3 29716 0

ETAPA 1
 AGORA = 105 EVENTO = 29702
 CRIEUM -3 1 631
 CONTTY = 4 VETYY(SETE) = 631
 SEJA 1 631 105
 COLOQUE 631 -2 29716
 COLOQUE 631 -2 29720
 AMOSTRADE 29686 1
 PROGRAME EVENTO = 29702 PARA AGORA + 1
 ETAPA 2

AGORA = 105 EVENTO = 29709
 TRANSFERENCIA 1 = 619
 REMOVA 619 29720 T
 DESTRUA 619
 I = 1 VETYY(SETE) = 615
 I = 2 VETYY(SETE) = 629
 I = 3 VETYY(SETE) = 631

ADICIONE 1 1 29789
 A 1 29789 3
 ETAPA 3
 AGORA = 105 ATIVIDADES CONDICIONAIS
 A 1 29789 3
 O -3 29716 1
 SUBTRAIA 1 1 29789
 A 1 29789 2
 RETIRE -1 29716 631
 A 1 631 105
 ANOTE 0 1 29793
 REGISTRE 0 29759
 AMOSTRADE 29694 6
 PROGRAME EVENTO = 29709 PARA AGORA + 6
 TRANSFERENCIA 1 = 631
 A 1 29789 2
 O -3 29716 0

```

ETAPA                2
AGORA = 106  EVENTO = 29709
  TRANSFERENCIA 1 = 629
  REMOVA                629      29720      T
  DESTRUA              629
I = 1    VETYY(SETE) = 615
I = 2    VETYY(SETE) = 631
  ADICIONE            1      1 29789
  A                  1 29789      3
  ETAPA              1
AGORA = 106  EVENTO = 29702
  CRIEUM             -3      1 647
  CONTYY = 3    VETYY(SETE) = 647
  SEJA              1      647 106
  COLOQUE           647     -2 29716
  COLOQUE           647     -2 29720
  AMOSTRADE        29686     11
  PROGRAME EVENTO = 29702  PARA AGORA + 11
  ETAPA              3
AGORA = 106  ATIVIDADES CONDICIONAIS
  A                  1 29789      3
  O                  -3 29716      1
  SUBTRAI A         1      1 29789
  A                  1 29789      2
  RETIRE            -1 29716     647
  A                  1      647 106
  ANOTE              0      1 29793
  REGISTRE           0 29759
  AMOSTRADE        29694      5
  PROGRAME EVENTO = 29709  PARA AGORA + 5
  TRANSFERENCIA 1 = 647
  A                  1 29789      2
  O                  -3 29716      0

```

-ENTRE COM A OPCAO DESEJADA - AGORA = 110

**-DESLIGAR

-QUE DESEJA DESLIGAR (CHAVES DO COMENTARIOS)

**-CHAVES

-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)

**-8

-ENTRE COM A OPCAO DESEJADA - AGORA = 110

**-CORRER

-QUANTO TEMPO

**- 90

-ENTRE COM A OPCAO DESEJADA - AGORA = 200

-*-RELATORIO

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)

-*- 1

-ENTRE COM O NOME DA ESTATISTICA

-*-FILA DE CAMINHOES

** IMPRESSAO DA ESTATISTICA FILA DE CAMINHOES
29724

AMOSTRA TAMANHO = 198 MENOR = 1 MAIOR = 9
MEDIA = 4.19 DESVIO PADRAO = 1.56

HISTOGRAMA

INTERVALO	FREQUENCIA	
(1)	0	I
(1, 2)	5	I*****
(2, 3)	11	I*****
(3, 4)	13	I*****
(4, 5)	28	I*****
(5, 6)	27	I*****
(6, 7)	9	I*****
(7, 8)	5	I*****
(8, 9)	2	I**
(9, 10)	1	I*
(10, 11)	0	I
(11, 12)	0	I
(12, 13)	0	I
(13, 14)	0	I
(14, 15)	0	I
(15, 16)	0	I
(16, 17)	0	I
(17, 18)	0	I
(18, 19)	0	I
(19,)	0	I

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)
 --*-- 2
 -ENTRE COM O NOME DA ESTATISTICA
 --*-TEMPOS DE ESPERA DOS CAMINHOS

** IMPRESSAO DA ESTATISTICA TEMPOS DE ESPERA DOS CAMINHOS
 29759

AMOSTRA TAMANHO = 70 MENOR = 0 MAIOR = 10
 MEDIA = 0.53 DESVIO PADRAO = 1.46

HISTOGRAMA

INTERVALO	FREQUENCIA
(0, 5)	99
(5, 10)	0
(10, 15)	1
(15, 20)	0
(20, 25)	0
(25, 30)	0
(30, 35)	0
(35, 40)	0
(40, 45)	0
(45, 50)	0
(50, 55)	0
(55,)	0

I*****

-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA - 0 PARA PARAR)
 --*-- 0
 -QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - 0 PARA PARAR)
 --*-- 0

-ENTRE COM A OPCAO DESEJADA - AGORA = 200
 --*-FINALIZAR

IV.4. CONSIDERAÇÕES FINAIS

Caso fôssemos apresentar todas as possíveis alternativas de uso dos procedimentos enumerados, provavelmente teríamos que editar um trabalho à parte, e que talvez não conseguíssemos esgotá-las, pois estas alternativas dependem da sequência de ocorrências das necessidades de um estudo de simulação, bem como do próprio problema analisado. É bom lembrar também que, apesar de estarmos apresentando estas alternativas separadamente, o uso dos procedimentos do sistema proposto é interativo, devendo ser usado como melhor convier ao usuário, o que lhe permitirá uma agilização no processo de análise de um modelo de simulação. Deste modo, acreditamos estar contribuindo para uma melhora sensível nos esforços despendidos na elaboração de um estudo de simulação, fazendo com que esta técnica seja ainda mais difundida e aplicada.

CAPÍTULO V

CONCLUSÕES FINAIS E SUGESTÕES

As maiores dificuldades até hoje encontradas pelos pesquisadores em simulação são:

a) - as diversas "visões-do-mundo": isto não tem permitido uma conceituação mais geral da técnica, impedindo a sua consolidação como ciência. Esta diversidade têm contribuído também para o aparecimento de inúmeros programas, linguagens, etc., cada uma com suas particularidades, restritas a um tipo de conceituação própria de quem as desenvolveu, tornando difícil a escolha por parte dos usuários. Existiam, até 1972, só nos Estados Unidos da América do Norte, cerca de 170 linguagens^{4,22} e muitas outras estão sendo desenvolvidas ainda hoje²⁸, contribuindo ainda mais para uma falsa orientação;

b) - o longo tempo e excessivos esforços despendidos no desenvolvimento de estudos de simulação: costumam, às vezes, ser abandonados no meio do projeto, devido à demora na apresentação dos resultados; é uma de suas facetas mais criticadas.

Esforços vêm sendo feitos para a redução ou eliminação destes inconvenientes. O recente trabalho de Antônio Jorge²² apresenta sugestões visando simplificar e unificar os conceitos de simulação. Nossa sugestão, em conjunto com os geradores^{14,15,21}, procura atenuar os inconvenientes da demora e dos grandes esforços no desenvolvimento de um projeto de simulação.

Apresenta, juntamente com o XLSIM2/FOR e o GERASIM, entre outras, as seguintes facilidades necessárias a um monitor interativo de simulação³⁵:

- 1) - impressões de gráficos;
- 2) - monitoração do estado do modelo durante a simulação;
- 3) - controle sobre o tempo simulado;
- 4) - habilidade para reinicializar a simulação;
- 5) - salvamento do estado do modelo a qualquer tempo simulado;
- 6) - modificação de características e parâmetros do modelo;
- 7) - coleta de medidas de desempenho do modelo;
- 8) - geração automática de dados;
- 9) - efetivo conjunto de diagnósticos de erros.

Deste modo, esperamos que lamentações do seguinte tipo: "modelos levam, geralmente, 3 a 4 vezes o tempo prometido para a sua elaboração"²⁶, não mais ocorram, bem como o argumento de que "o desenvolvimento de um bom modelo de simulação é sempre exaustivo e consome bastante tempo, requerendo alta habilidade e talento que podem não ser facilmente disponíveis"¹, deixe de ser apresentado como uma das desvantagens do uso de simulação.

Este sistema encontra-se, juntamente com a linguagem XLSIM2/FOR, disponível no programa de Administração (COPPEAD) da Coordenação dos Programas de Pós Graduação em Engenharia (COPPE) da Universidade Federal do Rio de Janeiro. Esta proposta, no entanto, não esgota este assunto. Detalhes adicionais devem ser acrescentados ao sistema, procurando cobrir uma gama maior de alternativas para a análise de resultados: esta parte é de fácil implementação, pois nosso sistema é dinâmico e permite a inclusão de outras facilidades de uma forma bastante simples.

Outra opção de pesquisa seria a unificação dos geradores com o sistema proposto, o que tornaria possível o desen-

volvimento das etapas de 1 a 8 (citadas no capítulo II) de forma interativa, permitindo maiores ganhos de tempo e esforços despendidos num estudo de simulação.

Também seria interessante a conjugação do sistema apresentado com alguns pacotes estatísticos, por exemplo, SPSS²⁷, de tal forma a permitir a execução de análises estatísticas (análise de variância, regressão, etc.) dos resultados, ainda de forma interativa, ganhando-se ainda mais em economia de tempo e esforços.

Deste modo, espera-se que a técnica de simulação apesar de bastante utilizada na solução de problemas, não seja usada como último recurso, mas sim como uma ciência com conceitos comuns e bem generalizados dentro da comunidade de Pesquisa Operacional, se não mundial, pelo menos nacional.

BIBLIOGRAFIA

1. SHANNON, R.E. - Systems simulation: the art and science. New Jersey, Prentice-Hall, 1975.
2. HULL, T.B. e DOBELL, A.R. - Random Number Generators. SIAM Review, 4(3), 1962.
3. JANSSON, B. - Random number generators. Stockholm, Almqvist Wiksell Publishers, 1966.
4. SHANNON, R.E. - Simulation: A Survey with Research Suggestions. AIIE Transactions, 7(3): 289-301, 1975.
5. KRASNOW, H.S. e MERIKALLIO, R.A. - The Past, Present and Future of General Simulation Languages. Management Science, 11(2): 236-267, 1964.
6. CARVALHO, R.S. e PONTES, A.J.X.C. - A Simulação no Contexto das Estruturas de Linguagens de Simulação. Rio de Janeiro, SOBRAPO - Anais do XII Simpósio Brasileiro de Pesquisa Operacional, Vol. II, 1979.
7. TOCHER, K.D. - The art of simulation. London, English Universities Press, 1963.
8. CARVALHO, R.S. - A System for Experimentation with Cellular Simulation Models. Ph.D. Thesis - University of Lancaster, Operational Research Department, 1975.
9. ——— e CROOKES, J.G. - Cellular Simulation. Operational Research Quartely, 27(1): 31-40, 1976.
10. MATHWESON, S.C. - Cellular Simulation: A Comment. Operational Research Quartely, 28(3), 1977.
11. GREENBERG, S. - GPSS primer. New York, John Wiley & Sons, Inc., 1972.

12. GPSS/360 - Introductory User's Manual. IBM, GH20-0304-4, 1969.
13. PRITSKER, A.A.B. - QGERTS Network Simulation for Anali-
sing Queueing and Logistics Problems. Software Packa-
ges Available from Pritsker & Associates, Inc.
14. MATHEWSON, S.C. - Simulation Program Generators. Simula-
tion, 23(6): 181-189, 1974.
15. CLEMENTSON, A.T. - Extended Control and Simulation Lan-
guage - Users Manual. University of Birmingham, The
Institute for Engineering Production, 1973.
16. NAYLOR, T.H. e FINGER, J.M. - Verification of Computer
Simulation Models. Management Science, 14(2), 1967.
17. PHILLIPS, D.T. - Applied Goodnes of Fit Testing. O.R. Mo-
nograph Series, AIIE-OR-72-1, American Institute of
Industrial Engineers, Inc., Georgia, 1972.
18. CONWAY, R.W. - Some Tactical Problems in Digital Simula-
tion. Management Science, 10(1): 47-61, 1963.
19. FARIA, F.F. - Simulação de Sistemas Discretos: Metodolo-
gia de Aplicação à Linguagem XLSIM. Tese de Mestrado
COPPEAD/UFRJ, Rio de Janeiro, 1978.
20. EHRENFELD, S.C. S.BEN.TUVIA - The Efficiency of Statisti-
cal Simulation Procedures. Technometrics, 4(2), 1952.
21. PONTES, A.J.X.C. - Metodologia para Geração de Programas
em Simulação Celular. Tese de Doutorado - COPPE/UFRJ,
Rio de Janeiro, 1981.
22. SAMMET, J.E. - Programming Languages: History and Future.
Communications of the ACM, 15(7), 1972.

23. CARVALHO, R.S. e FARIA, F.F. - Manual do XLSIM2: Versão FORTRAN. Relatório Interno, COPPEAD/UFRJ, Rio de Janeiro, 1978.
24. PRITSKER, A.A.B. - The GASP IV simulation language. New York, John Wiley & Sons, 1974.
25. KIVIAT, P.J. et alii - The SIMSCRIPT II programming language. New Jersey, Prentice-Hall, 1968.
26. GOLDIE, J.H. - Simulation and Irritation. Supplement to Corporate Simulation Models - Graduate School of Business Administration of the University of Washington, Washington, 1970.
27. NIE, N.H. et alii - Statistical package for the social sciences: SPSS. New York, McGraw-Hill, 1975.
28. UYENO, D.H. - PASSIM: A Discret-Event Simulation Package for PASCAL. Simulation, 35(6): 183-190, 1980.
29. NAYLOR, T.H. et alii - Computer simulation techniques. New York, John Wiley & Sons, 1968.
30. GORDON, G. - System simulation. New Jersey, Prentice-Hall 1969.
31. RODRIGUES Filho, V. et alii - Controle de Combustíveis numa Usina Siderúrgica: Modelo Matemático e Simulação Rio de Janeiro, SOBRAPO - Anais do VII Simpósio Brasileiro de Pesquisa Operacional, Vol. I, 1974.
32. ——— e COSTA, H.F. - Simulação do Sistema de Reparos de Torpedos da USIMINAS. Rio de Janeiro, SOBRAPO - Anais do X Simpósio Brasileiro de Pesquisa Operacional, Vol. 2, 1971.

33. ——— e Oliveira, J.M.C. - Simulação do Transporte Ferroviário de Placas da Aciaria (Lingotamento Contínuo) para a Laminação. Revista Brasileira de Pesquisa Operacional, 3(1), 1980.
34. ——— et alii - Análise das Alternativas de Produção da Aciaria Nº 1 da USIMINAS com a Operação da Estação de Desgaseificação: Uma Aplicação da Simulação. Salvador, COAÇO/COREF/ABM - Anais do Seminário sobre Aciaria e Refratários, 1980.
35. SOHNLE, R.C. et alii - Requirements for Interactive Simulation Systems. Simulation, 20(5): 145-152, 1973.
36. CARVALHO, R.S. e FARIA, F.F. - Simulação com XLSIM2 - Versão FORTRAN. São Paulo, SOBRAPO - Anais do XII Simpósio Brasileiro de Pesquisa Operacional, Vol.II, 1979.

APÊNDICE I

SISTEMA ANALISADOR

** LISTAGEM DO PROGRAMA
DE COMPUTADOR **

SUBROUTINE ANAYY

129

REAL MEDIA

LOGICAL EXIST,LEIA,RMOVA

INTEGER O,A,AMSTA

INTEGER YY(30000),VETYY(10,50),CONTTY(10)

INTEGER TRNSF(20),CHAVE(7),VALOR(40),CELLA(50),FREQC(40),

*ENTDD,TRSAO,CONJT,PONTR,PRMRO,ULTMO,PDSOR,SUCSR,ELMTO,

*TAMAN,ENTRA,SAIDA,AGORA,ERLNG,UNFRM,EMPCA,YYB,YYE,

*CRESC,DECRS,ATIVA,DIFTE,GLOBAL,UM,DOIS,TRES,QUATR,CINCO,

*SEIS,SETE,OITO,NOVE,DEZ,ONZE,DOZE,TREZE,ZERO,TSTYY,TERM

DIMENSION YYDP(100),TITLO(10)

COMMON IMPR,LEIT,TRNSF,CHAVE,VALOR,CELLA,FREQC,TITLO,

*INFTO,AGORA,UM,DOIS,TRES,QUATR,CINCO,SEIS,SETE

*,OITO,NOVE,DEZ,ONZE,DOZE,TREZE,ZERO,NINTG,DECRS,INTGR,

*ENTRA,CRESC,ATIVA,INTNO,GLOBAL,IGUAL,NNHUM,CONJT,SAIDA,

*PRMRO,NRMAL,DIFTE,ENTDD,ULTMO,ERLNG,TRSAO,PDSOR,TAMAN,

*UNFRM,PONTR,SUCSR,EMPCA,ELMTO,MAXMO,MINMO

COMMON IFLA,IFLB,YYB,YYE,NYY,TSTYY,JKYY,YYDP,NYYDP,YY

COMMON AGOYY,KKYY,JJYY,TERM,VETYY,CONTTY

INTEGER QUEST(3)

DIMENSION KFLAG(3),IDENT(36),INOME(2),ICOD(26),IFLAG(5)

DIMENSION INFORM(20)

DATA ICOD/'CORR','LIGA','DESL','TIRA','LIMP','RELA','AUTO','REST',

*'TEST','ALTE','MODI','REPR','ATIV','DESA','TROC','ANOT','DEFI',

*'MUDA','REIN','GERA','CANC','PROC','LIST','PERM','ORDE','DEST'/

DATA QUEST/'SIM','NAO','FIM'/

DATA IFIM/'FINA'/

DATA IDENT/'ENTI','TRAN','RETI','COLO','PRIM','ULTI','QUAL','PROG'

*, 'APAG','NENH','ESTA','DIST','NORM','ERLA','UNIF','EMPI','MEDI',

*'MAXI','MINI','DESV','TAMA','CHAV','COME','HIST','GRUP','CONJ','OU

*TR','DETE','TODO','IGUA','DIFE','MAIO','MENO','PONT','CRES','DECR'

*/

DATA JTERM/6/

5 WRITE(JTERM,7)

7 FORMAT(1H,///)

IF(INFORM(1).EQ.ZERO) GO TO 12

IF(INFORM(3).EQ.1) GO TO 15

WRITE(JTERM,10)

10 FORMAT(1H,/,1H,11X,'S I S T E M A A N A L I S A D O R',//,

*1H,20X,'OPCOES OFERECIDAS',//,

*1H,11X,' - CORRER O MODELO' //,

*1H,11X,' - LIGAR CHAVES OU COMENTARIOS' //,

*1H,11X,' - DESLIGAR CHAVES OU COMENTARIOS' //,

*1H,11X,' - TIRAR UMA FOTOGRAFIA DAS CONDICOES ATUAIS DA COR-' //,

*1H,11X,' - RIDA' //,

*1H,11X,' - LIMPEZA DE ESTATISTICAS E/OU HISTORICOS' //,

*1H,11X,' - RELATORIO(IMPRESSAO) DE ESTATISTICAS' //,

*1H,11X,' E/OU GRAFICOS' //,

*1H,11X,' - AUTOCORRELACAO(CORRELOGRAMAS E/OU' //,

*1H,11X,' CORRELOBLOCOS)' //,

*1H,11X,' - RESTABELECEER AS CONDICOES INICIAIS DESTA CORRIDA' //,

*1H,11X,' - TESTES DE KOLMOGOROV-SMIRNOV' //,

*1H,11X,' - ALTERAR ATRIBUTO DE ENTIDADE' //,

*1H,11X,' - MODIFICAR CONJUNTOS(RETIRAR OU COLOCAR ELEMENTOS' //,

*1H,11X,' - REPROGRAMAR EVENTOS(PROGRAMAR NOVOS OU APAGAR' //,

*1H,11X,' EVENTOS JA PROGRAMADOS)' //,

WRITE(JTERM,1111)

```

1111 FORMAT(1H ,11X, ' - ATIVAR CELULAS' ,/,
*1H ,11X, ' - DESATIVAR CELULAS' ,/,
*1H ,11X, ' - TROCAR PARAMETROS DE DISTRIBUICOES' ,/,
*1H ,11X, ' - ANOTAR, EM HISTORICOS, PARAMETROS(MEDIA,MAXIMO,MI- ,/,
*1H ,11X, ' NIMO, DESVIO PADRAO, TAMANHO) DAS ESTATISTICAS' ,/,
*1H ,11X, ' USADAS' ,/,
*1H ,11X, ' - DEFINIR HISTORICOS' ,/,
*1H ,11X, ' - MUDAR A SEMENTE DA SEQUENCIA DE VALORES ALEATORIOS' ,/,
*1H ,11X, ' - REINICIALIZAR A SEMENTE DA SEQUENCIA DE VALORES' ,/,
*1H ,11X, ' ALEATORIOS' ,/,
*1H ,11X, ' - GERAR VARIAVEIS ANTITETICAS' ,/,
*1H ,11X, ' - CANCELAR A GERACAO DE VARIAVEIS ANTITETICAS')
WRITE(JTERM,1112)
1112 FORMAT(1H ,11X, ' - PROCURAR ELEMENTOS EM CONJUNTOS OU GRUPOS' ,/,
*1H ,11X, ' - LISTAR VALOR(ES) DE ATRIBUTO(S) DE ELEMENTO(S)' ,/,
*1H ,11X, ' - PERMUTAR POSICAO DE ELEMENTO DENTRO DE CONJUNTO' ,/,
*1H ,11X, ' - ORDENAR CONJUNTO(S)' ,/,
*1H ,11X, ' - DESTRUIR ELEMENTO(S) CRIADO(S)' ,/,
*1H ,11X, ' - FINALIZAR A ANALISE',/)
GO TO 15
12 WRITE(JTERM,13)
13 FORMAT(1H ,/,1H ,11X, 'S I S T E M A A N A L I S A D O R',/,/,
*1H ,13X, 'OPCOES OFERECIDAS NESTE MOMENTO',/,/,
*1H ,11X, ' - CORRER O MODELO' ,/,
*1H ,11X, ' - LIGAR CHAVES OU COMENTARIOS' ,/,
*1H ,11X, ' - DESLIGAR CHAVES OU COMENTARIOS',/)
INFORM(1)=1
AGOYY=ZERO
15 WRITE(JTERM,16) AGORA
16 FORMAT(1H , ' -ENTRE COM A OPCAO DESEJADA - AGORA = ',I6,/)
IF(INFORM(3).EQ.0) WRITE(JTERM,116)
116 FORMAT(1H ,
*1H , ' *-OBS. NAO E NECESSARIO DIGITAR TODA A DESCRICAO',/,
*1H , ' DA OPCAO DESEJADA, BASTANDO APENAS DIGITAR',/,
*1H , ' A PRIMEIRA PALAVRA.',/)
17 READ(TERM,20) IFLAG
20 FORMAT(5A4)
WRITE(JTERM,25) IFLAG
25 FORMAT(1H , ' *- ',5A4,/)
IF(IFLAG(1).EQ.IFIM) CALL EXIT
NUMERO = 0
DO 30 I=1,26
NUMERO = I
IF(IFLAG(1).EQ.ICOD(I)) GO TO 50
30 CONTINUE
WRITE(JTERM,40) IFLAG
40 FORMAT(1H , ' -OPCAO ',5A4, ' INCORRETA.',/,1H ' -ENTRE COM A OPCAO CO
*RRRETA NOVAMENTE.',/)
GO TO 17
C
50 CONTINUE
GO TO(100,200,300,400,500,600,700,800,900,1000,2000,3000,4000,5000
*,6000,7000,8000,9000,10000,11000,12000,13000,14000,15000,16000,170
*00),NUMERO
C
100 CONTINUE
WRITE(JTERM,110)
110 FORMAT(1H , ' -QUANTO TEMPO ',/)
READ(TERM,120) ITEMPO

```

```

120 FORMAT(I6)
    AGOYY = AGORA + ITEMPO
    WRITE(JTERM,130) ITEMPO
130 FORMAT(1H , *-*, I6, /)
    RETURN

```

C

```

200 CONTINUE
    WRITE(JTERM,201)
201 FORMAT(1H , *-QUE DESEJA LIGAR (CHAVES OU COMENTARIOS)*, /)
202 READ(TERM,1020) KFLAG
    WRITE(JTERM,1030) KFLAG
    IF(KFLAG(1).EQ.IDENT(22)) GO TO 205
    IF(KFLAG(1).EQ.IDENT(23)) GO TO 250
    WRITE(JTERM,1040) KFLAG
    GO TO 202
205 WRITE(JTERM,210)
210 FORMAT(1H , *-QUAL CHAVE (8 PARA TODAS E 0 PARA PARAR)*, /)
    READ(TERM,220) ICHAV
220 FORMAT(I1)
    WRITE(JTERM,225) ICHAV
225 FORMAT(1H , *-*, I1, /)
    IF(ICHAV.EQ.ZERO) GO TO 5
    IF(ICHAV.EQ.OITO) GO TO 230
    CHAVE(ICHAV) = AGORA
    GO TO 200
230 DO 240 I=1,7
240 CHAVE(I) = AGORA
    GO TO 5
250 INFORM(3)=0
    GO TO 5

```

C

```

300 CONTINUE
    WRITE(JTERM,301)
301 FORMAT(1H , *-QUE DESEJA DESLIGAR (CHAVES OU COMENTARIOS)*, /)
302 READ(TERM,1020) KFLAG
    WRITE(JTERM,1030) KFLAG
    IF(KFLAG(1).EQ.IDENT(22)) GO TO 305
    IF(KFLAG(1).EQ.IDENT(23)) GO TO 330
    WRITE(JTERM,1040) KFLAG
    GO TO 302
305 WRITE(JTERM,210)
    READ(TERM,220) ICHAV
    WRITE(JTERM,225) ICHAV
    IF(ICHAV.EQ.ZERO) GO TO 5
    IF(ICHAV.EQ.OITO) GO TO 310
    CHAVE(ICHAV) = INFTO
    GO TO 305
310 DO 320 I=1,7
320 CHAVE(I) = INFTO
    GO TO 5
330 INFORM(3)=1
    GO TO 5

```

C

```

400 CONTINUE
410 FORMAT(1H , *-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA)*
    *, /)
420 FORMAT(I2)
425 FORMAT(1H , *-*, I2, /)
440 FORMAT(1H , *-QUAL HISTORICO SERA USADO (DE A ORDEM DE DEFINICAO DO

```

```
* MESMO) *,/)
INFORM(4)=1
CALL CMECE
GO TO 5
```

C

```
500 WRITE(JTERM,510)
510 FORMAT(1H , *-QUAL ESTATISTICA (DE A ORDEM DE DEFINICAO DA MESMA
*- O PARA PARAR) *,/)
READ(TERM,420) NUMEST
WRITE(JTERM,425) NUMEST
IF(NUMEST.EQ.ZERO) GO TO 520
CALL LIMPE(VETYY(1,NUMEST))
GO TO 500
520 WRITE(JTERM,530)
530 FORMAT(1H , *-QUAL HISTORICO (DE A ORDEM DE DEFINICAO DO MESMO - C
* PARA PARAR) *,/)
READ(TERM,420) NUMHIS
WRITE(JTERM,425) NUMHIS
IF(NUMHIS.EQ.ZERO) GO TO 5
CALL LIMPE(VETYY(4,NUMHIS))
GO TO 520
```

C

```
600 WRITE(JTERM,510)
READ(TERM,420) NUMEST
WRITE(JTERM,425) NUMEST
IF(NUMEST.EQ.ZERO) GO TO 610
608 WRITE(JTERM,602)
602 FORMAT(1H , *-ENTRE COM O NOME DA ESTATISTICA *,/)
READ(TERM,603) TITLO
603 FORMAT(10A4)
WRITE(JTERM,604) TITLO
604 FORMAT(1H , *-*-*,10A4,/)
CALL IMPMA(VETYY(1,NUMEST))
WRITE(JTERM,7)
GO TO 600
610 WRITE(JTERM,530)
READ(TERM,420) NUMHIS
WRITE(JTERM,425) NUMHIS
IF(NUMHIS.EQ.ZERO) GO TO 5
WRITE(JTERM,620)
620 FORMAT(1H , *-VALORES AGRUPADOS EM NUMERO NAO SUPERIOR A *,/)
READ(TERM,630) NDADOS
630 FORMAT(I3)
WRITE(JTERM,640) NDADOS
640 FORMAT(1H , *-*-*,I3,/)
CALL GRFCO(VETYY(4,NUMHIS),NDADOS)
WRITE(JTERM,7)
GO TO 610
```

C

```
700 WRITE(JTERM,710)
710 FORMAT(1H , *-DESEJA CORRELOGRAMA (SIM OU NAO) *,/)
715 READ(TERM,720) IPERG
720 FORMAT(A3)
WRITE(JTERM,722) IPERG
722 FORMAT(1H , *-*-*,A3,/)
IF(IPERG.EQ.QUEST(1)) GO TO 730
IF(IPERG.EQ.QUEST(2)) GO TO 770
WRITE(JTERM,725)
725 FORMAT(1H , *-CODIGO INVALIDO. DIGITE SIM OU NAO *,/)
```

```

GO TO 715
730 WRITE(JTERM,740)
740 FORMAT(1H , '-QUAL SERIE ',/)
    READ(TERM,220) NSER
    WRITE(JTERM,225) NSER
    WRITE(JTERM,440)
    READ(TERM,420) NUMHIS
    WRITE(JTERM,425) NUMHIS
    WRITE(JTERM,760)
760 FORMAT(1H , '-QUAL LAG MAXIMO ',/)
    READ(TERM,420) LAG
    WRITE(JTERM,425) LAG
    CALL CORGM(VETYY(4,NUMHIS),NSER,LAG)
    WRITE(JTERM,7)
    GO TO 700
770 WRITE(JTERM,780)
780 FORMAT(1H , 'DESEJA CORRELOBLOCO (SIM OU NAO)',/)
785 READ(TERM,720) IPERG
    WRITE(JTERM,722) IPERG
    IF(IPERG.EQ.QUEST(1)) GO TO 790
    IF(IPERG.EQ.QUEST(2)) GO TO 5
    WRITE(JTERM,725)
    GO TO 785
790 WRITE(JTERM,740)
    READ(TERM,220) NSER
    WRITE(JTERM,225) NSER
    WRITE(JTERM,440)
    READ(TERM,420) NUMHIS
    WRITE(JTERM,425) NUMHIS
    WRITE(JTERM,795)
795 FORMAT(1H , '-QUAL O NUMERO MAXIMO DE OBSERVACOES POR BLOCO ',/)
    READ(TERM,630) NDADOS
    WRITE(JTERM(640) NDADOS)
    CALL CRBLC(VETYY(4,NUMHIS),NSER,NDADOS)
    WRITE(JTERM,7)
    GO TO 770

C
800 CONTINUE
    INFORM(8)=1
    CALL RCMCE
    GO TO 5

C
900 WRITE(JTERM,510)
    READ(TERM,420) NUMEST
    WRITE(JTERM,425) NUMEST
    IF(NUMEST.EQ.ZERO) GO TO 5
    I = 0
905 WRITE(JTERM,910)
910 FORMAT(1H , '-QUAL E A FREQUENCIA (0 PARA PARAR)',/)
    READ(TERM,420) IFRE
    WRITE(JTERM,425) IFRE
    IF(IFRE.EQ.ZERO) GO TO 920
    I = I + 1
    FREQC(I) = IFRE
    GO TO 905
920 CALL KOLMO(VETYY(1,NUMEST),RX)
    WRITE(JTERM,7)
    GO TO 900

C

```

```

1000 CONTINUE
1005 WRITE(JTERM,1010)
1010 FORMAT(1H , *-QUE ELEMENTO DESEJA ALTERAR (ENTIDADE, TRANSACAO OU N
*ENHUM)*,/)
1015 READ(TERM,1020) KFLAG
1020 FORMAT(3A4)
WRITE(JTERM,1030) KFLAG
1030 FORMAT(1H , *-*-*,3A4,/)
IF(KFLAG(1).EQ.IDENT(1)) GO TO 1050
IF(KFLAG(1).EQ.IDENT(2)) GO TO 1110
IF(KFLAG(1).EQ.IDENT(10)) GO TO 5
WRITE(JTERM,1040) KFLAG
1040 FORMAT(1H , *-OPCAO *,3A4,* INVALIDA. DIGITE NOVAMENTE*,/)
GO TO 1015
1050 WRITE(JTERM,1060)
1060 FORMAT(1H , *-QUAL ENTIDADE (DE A ORDEM DE CRIACAO DA MESMA - O PA
*RA PARAR*,/)
READ(TERM,420) IORDEM
WRITE(JTERM,425) IORDEM
IF(IORDEM.EQ.ZERO) GO TO 1005
IELEM = VETYY(5,IORDEM)
1065 WRITE(JTERM,1070)
1070 FORMAT(1H , *-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO - O
*PARA PARAR)*,/)
READ(TERM,420) NATRIB
WRITE(JTERM,425) NATRIB
IF(NATRIB.EQ.ZERO) GO TO 1050
1090 WRITE(JTERM,1100)
1100 FORMAT(1H , *-QUAL O NOVO VALOR DO ATRIBUTO *,/)
READ(TERM,120) IVALOR
WRITE(JTERM,130) IVALOR
CALL SEJA(NATRIB,IELEM,IVALOR)
GO TO 1065
1110 WRITE(JTERM,1120)
1120 FORMAT(1H , *-QUAL TRANSACAO (DE A ORDEM DE DEFINICAO DA MESMA - O
* PARA PARAR)*,/)
READ(TERM,420) IORDEM
WRITE(JTERM,425) IORDEM
IF(IORDEM.EQ.ZERO) GO TO 1005
IELEM = VETYY(SETTE,IORDEM)
1130 WRITE(JTERM,1070)
READ(TERM,420) NATRIB
WRITE(JTERM,425) NATRIB
IF(NATRIB.EQ.ZERO) GO TO 1110
WRITE(JTERM,1100)
READ(TERM,120) IVALOR
WRITE(JTERM,130) IVALOR
CALL SEJA(NATRIB,IELEM,IVALOR)
GO TO 1130
C
2000 CONTINUE
2002 WRITE(JTERM,2005)
2005 FORMAT(1H , *-QUAL CONJUNTO (DE A ORDEM DE DEFINICAO DO MESMO - O
*PARA PARAR)*,/)
READ(TERM,420) NUMERO
WRITE(JTERM,425) NUMERO
IF(NUMERO.EQ.ZERO) GO TO 5
ICONJ = VETYY(6,NUMERO)
WRITE(JTERM,2010)

```

```

2010 FORMAT(1H , *-QUAL MODIFICACAO (RETIRAR OU COLOCAR)*,/)
2015 READ(TERM,2020) INOME
2020 FORMAT(2A4)
WRITE(JTERM,2030) INOME
2030 FORMAT(1H , *-*-*,2A4,/)
IF(INOME(1).EQ.IDENT(3)) GO TO 2050
IF(INOME(1).EQ.IDENT(4)) GO TO 2170
WRITE(JTERM,2040) INOME
2040 FORMAT(1H , *-OPCAO *,2A4,* INVALIDA. DIGITE NOVAMENTE*,/)
GO TO 2015
2050 WRITE(JTERM,2060)
2060 FORMAT(1H , *-QUAL ELEMENTO (PRIMEIRO,ULTIMO,QUALQUER,DETERMINADO(
*S) OU TODOS)*,/)
2070 READ(TERM,1020) KFLAG
WRITE(JTERM,1030) KFLAG
IF(KFLAG(1).EQ.IDENT(5)) GO TO 2080
IF(KFLAG(1).EQ.IDENT(6)) GO TO 2090
IF(KFLAG(1).EQ.IDENT(7)) GO TO 2100
IF(KFLAG(1).EQ.IDENT(28)) GO TO 2300
IF(KFLAG(1).EQ.IDENT(29)) GO TO 2500
WRITE(JTERM,2040) INOME
GO TO 2070
2080 CALL RTIRE(PRMRO,ICONJ,ISUMA)
GO TO 2002
2090 CALL RTIRE(ULTMO,ICONJ,ISUMA)
GO TO 2002
2100 WRITE(JTERM,2110)
2110 FORMAT(1H , *-QUE TIPO DE ELEMENTO (ENTIDADE OU TRANSACAO)*,/)
2120 READ(TERM,1020) KFLAG
WRITE(JTERM,1030) KFLAG
IF(KFLAG(1).EQ.IDENT(1)) GO TO 2130
IF(KFLAG(1).EQ.IDENT(2)) GO TO 2140
WRITE(JTERM,1040) KFLAG
GO TO 2120
2130 INDICE = CINCO
GO TO 2150
2140 INDICE = SETE
2150 WRITE(JTERM,2160)
2160 FORMAT(1H , *-QUAL ELEMENTO (DE A ORDEM DE DEFINICAO DO MESMO)*,/)
READ(TERM,420) INUM
WRITE(JTERM,425) INUM
IELEM = VETYY(INDICE,INUM)
IF(.NOT.RMOVA(IELEM,ICONJ)) GO TO 2161
GO TO 2000
2161 WRITE(JTERM,2162) KFLAG, INUM, NUMERO
2162 FORMAT(1H , *-NAO EXISTE A *,3A4,* DE ORDEM *,I2,* NO CONJUNTO DE O
*RDEM *,I2,**,/,1H , *-ALGUM ERRO DE DIGITACAO. REPITA TUDO NOVAMEN
*TE*,/)
GO TO 2002
2170 WRITE(JTERM,2110)
2180 READ(TERM,1020) KFLAG
WRITE(JTERM,1030) KFLAG
IF(KFLAG(1).EQ.IDENT(1)) GO TO 2190
IF(KFLAG(1).EQ.IDENT(2)) GO TO 2200
WRITE(JTERM,1040) KFLAG
GO TO 2180
2190 INDICE = CINCO
GO TO 2210
2200 INDICE = SETE

```

```

2210 WRITE(JTERM,2160)
      READ(TERM,420) INUM
      WRITE(JTERM,425) INUM
      IELEM = VETYY(INDICE,INUM)
      WRITE(JTERM,2220)
2220 FORMAT(1H , *-QUAL POSICAO (PRIMEIRO OU ULTIMO)*,/)
2230 READ(TERM,2020) INOME
      WRITE(JTERM,2030) INOME
      IF(INOME(1).EQ.IDENT(5)) GO TO 2240
      IF(INOME(1).EQ.IDENT(6)) GO TO 2250
      WRITE(JTERM,2040) INOME
      GO TO 2230
2240 CALL CLQUE(IELEM,PRMO,ICONJ)
      GO TO 2002
2250 CALL CLQUE(IELEM,ULTMO,ICONJ)
      GO TO 2002
2300 WRITE(JTERM,13050)
2310 READ(TERM,20) IFLAG
      WRITE(JTERM,25) IFLAG
      IF(IFLAG(1).EQ.IDENT(21)) GO TO 2320
      IF(IFLAG(1).EQ.IDENT(27)) GO TO 2330
      WRITE(JTERM,40) IFLAG
      GO TO 2310
2320 NATRIB=-3
      GO TO 2340
2330 WRITE(JTERM,13090)
      READ(TERM,420) NATRIB
      WRITE(JTERM,425) NATRIB
2340 WRITE(JTERM,2350)
2350 FORMAT(1H , *-O(S) ELEMENTO(S) RETIRADO(S) SERA(O) *,/,1H ,*-(IGUAL
*, DIFERENTE, MAIOR QUE OU MENOR QUE)*,/)
2360 READ(TERM,20) IFLAG
      WRITE(JTERM,25) IFLAG
      IF(IFLAG(1).EQ.IDENT(30)) GO TO 2370
      IF(IFLAG(1).EQ.IDENT(31)) GO TO 2380
      IF(IFLAG(1).EQ.IDENT(32)) GO TO 2390
      IF(IFLAG(1).EQ.IDENT(33)) GO TO 2400
      WRITE(JTERM,40) IFLAG
      GO TO 2360
2370 ITIPO=-1
      GO TO 2410
2380 ITIPO=0
      GO TO 2410
2390 ITIPO=INFTO
      GO TO 2410
2400 ITIPO=-INFTO
2410 WRITE(JTERM,2420)
2420 FORMAT(1H , *-DE QUAL VALOR *,/)
      READ(TERM,120) IVALOR
      WRITE(JTERM,130) IVALOR
      CALL RTNJA(ICONJ,NATRIB,ITIPO,IVALOR)
      GO TO 2002
2500 CALL Esvaz(ICONJ)
      GO TO 2002
C
3000 CONTINUE
3005 WRITE(JTERM,3010)
3010 FORMAT(1H , *-QUE REPROGRAMACAO DESEJA (PROGRAMAR OU APAGAR EVENTO
*S OU NENHUMA DELAS)*,/)

```

```

3020 READ(TERM,1020) KFLAG
      WRITE(JTERM,1030) KFLAG
      IF(KFLAG(1).EQ.IDENT(8)) GO TO 3030
      IF(KFLAG(1).EQ.IDENT(9)) GO TO 3070
      IF(KFLAG(1).EQ.IDENT(10)) GO TO 5
      WRITE(JTERM,1040) KFLAG
      GO TO 3020
3030 WRITE(JTERM,3040)
3040 FORMAT(1H , '-QUAL EVENTO (DE A ORDEM DE DEFINICAO DO MESMO)',/)
      READ(TERM,420) NUMEVE
      WRITE(JTERM,425) NUMEVE
      IEVENT = VETYY(3,NUMEVE)
      WRITE(JTERM,3050)
3050 FORMAT(1H , '-QUAL O TEMPO ',/)
      READ(TERM,120) ITEMPO
      WRITE(JTERM,130) ITEMPO
3051 WRITE(JTERM,3052)
3052 FORMAT(1H , '-ALGUM ELEMENTO (ENTIDADE OU TRANSACAO) TRANSFERIDO NE
*STE EVENTO ',/,1H , '-(SIM, NAO OU FIM)',/)
3053 READ(TERM,720) IPERG
      WRITE(JTERM,722) IPERG
      IF(IPERG.EQ.QUEST(1)) GO TO 3054
      IF(IPERG.EQ.QUEST(2)) GO TO 3061
      IF(IPERG.EQ.QUEST(3)) GO TO 3060
      WRITE(JTERM,3100)
      GO TO 3053
3054 WRITE(JTERM,2110)
3055 READ(TERM,1020) KFLAG
      WRITE(JTERM,1030) KFLAG
      IF(KFLAG(1).EQ.IDENT(1)) GO TO 3056
      IF(KFLAG(1).EQ.IDENT(2)) GO TO 3057
      WRITE(JTERM,1040) KFLAG
      GO TO 3055
3056 INDICE = CINCO
      GO TO 3058
3057 INDICE = SETE
3058 WRITE(JTERM,2160)
      READ(TERM,420) INUM
      WRITE(JTERM,425) INUM
      IELEM = VETYY(INDICE,INUM)
      WRITE(JTERM,3059)
3059 FORMAT(1H , '-QUAL A ORDEM DE TRANSFERENCIA ',/)
      READ(TERM,420) IND
      WRITE(JTERM,425) IND
      TRNSF(IND) = IELEM
      GO TO 3051
3060 CALL PRGME(IEVENT,ITEMPO)
      GO TO 3005
3061 CALL PRGME(IEVENT,ITEMPO)
      GO TO 3005
3070 WRITE(JTERM,3040)
      READ(TERM,420) NUMEVE
      WRITE(JTERM,425) NUMEVE
      IEVENT = VETYY(3,NUMEVE)
3080 WRITE(JTERM,3085)
3085 FORMAT(1H , '-ALGUM ELEMENTO (ENTIDADE OU TRANSACAO) TRANSFERIDO NE
*STE EVENTO ',/,1H , '-(SIM OU NAO)',/)
3090 READ(TERM,720) IPERG
      WRITE(JTERM,722) IPERG

```

```

IF(IPERG.EQ.QUEST(1)) GO TO 3110
IF(IPERG.EQ.QUEST(2)) GO TO 3220
WRITE(JTERM,725)
3100 FORMAT(1H , *-CODIGO INVALIDO. DIGITE SIM, NAO OU FIM. */,)
GO TO 3090
3110 WRITE(JTERM,2110)
3120 READ(TERM,1020) KFLAG
WRITE(JTERM,1030) KFLAG
IF(KFLAG(1).EQ.IDENT(1)) GO TO 3130
IF(KFLAG(1).EQ.IDENT(2)) GO TO 3140
WRITE(JTERM,1040) KFLAG
GO TO 3120
3130 INDICE = CINCO
GO TO 3150
3140 INDICE = SETE
3150 WRITE(JTERM,2160)
READ(TERM,420) INUM
WRITE(JTERM,425) INUM
IELEM = VETYY(INDICE,INUM)
WRITE(JTERM,3155)
3155 FORMAT(1H , *-QUAL FOI A ORDEM DE TRANSFERENCIA DESTE ELEMENTO */,)
READ(TERM,420) IORD
WRITE(JTERM,425) IORD
3160 I = 0
3170 I = I + 1
IF(.NOT.LEIA(I,IEV,ITEMPO,TRNSF)) GO TO 3200
IF(IEV.NE.IEVENT) GO TO 3170
IF(TRNSF(IORD).EQ.IELEM) GO TO 3190
WRITE(JTERM,3180) KFLAG, INUM, IORD
3180 FORMAT(1H , *-A ',3A4,' DE NUMERO ',I2,' E DE ORDEM ',I2,' DE TRANS
*FERENCIA',/,1H , *-NAO FOI ENCONTRADA. ALGUM ERRO OCORREU. FAVOR RE
*PETIR OS PASSOS NOVAMENTE. */,)
GO TO 3110
3190 CALL APAGG(I,IEV,ITEMPO,TRNSF)
GO TO 3005
3200 WRITE(JTERM,3210) NUMEVE
3210 FORMAT(1H , *-O EVENTO DE NUMERO ',I2,' AINDA NAO FOI PROGRAMADO.',
*/,1H , *-ALGUM ERRO DE DIGITACAO. FAVOR REPETIR OS PASSOS NOVAMENTE
*.',/,)
GO TO 3070
3220 I = 0
3230 I = I + 1
IF(.NOT.LEIA(I,IEV,ITEMPO,TRNSF)) GO TO 3240
IF(IEV.NE.IEVENT) GO TO 3230
CALL APAGG(I,IEV,ITEMPO,TRNSF)
GO TO 3005
3240 WRITE(JTERM,3210) NUMEVE
GO TO 3070
C
4000 CONTINUE
4005 WRITE(JTERM,4010)
4010 FORMAT(1H , *-QUAL CELULA (0 PARA PARAR)',/,)
READ(TERM,420) ICEL
WRITE(JTERM,425) ICEL
IF(ICEL.EQ.ZERO) GO TO 5
CELLA(ICEL)=ATIVA
GO TO 4005
C
5000 CONTINUE

```

CALL DTIVE
GO TO 5

C

```

6000 CONTINUE
6200 WRITE(JTERM,6210)
6210 FORMAT(1H , *-QUAL DISTRIBUICAO (DE A ORDEM DE DEFINICAO DA MESMA
*- O PARA PARAR) *,/)
    READ(TERM,420) NDIST
    WRITE(JTERM(425) NDIST
    IF(NDIST.EQ.ZERO) GO TO 5
    WRITE(JTERM,6220)
6220 FORMAT(1H , *-QUAL O TIPO DA DISTRIBUICAO (NORMAL, ERLANG, UNIFORM
*E OU EMPIRICA) *,/)
6230 READ(TERM,2020) INOME
    WRITE(JTERM,2030) INOME
    IF(INOME(1).EQ.IDENT(13)) GO TO 6240
    IF(INOME(1).EQ.IDENT(14)) GO TO 6280
    IF(INOME(1).EQ.IDENT(15)) GO TO 6400
    IF(INOME(1).EQ.IDENT(16)) GO TO 6500
    WRITE(JTERM,2040)
    GO TO 6230
6240 WRITE(JTERM,6250)
6250 FORMAT(1H , *-QUAL A MEDIA DA NORMAL *,/)
    READ(TERM,120) NMED
    WRITE(JTERM,130) NMED
    WRITE(JTERM,6260)
6260 FORMAT(1H , *-QUAL O DESVIO PADRAO DA NORMAL *,/)
    READ(TERM,120) NDESV
    WRITE(JTERM,130) NDESV
    CALL DSTRB(VETYY(2,NDIST),NRMAL,NMED,NDESV,0)
    GO TO 6200
6280 WRITE(JTERM,6290)
6290 FORMAT(1H , *-QUAL A MEDIA DE CADA ESTAGIO DA ERLANG *,/)
    READ(TERM,120) NMED
    WRITE(JTERM,130) NMED
    WRITE(JTERM,6300)
6300 FORMAT(1H , *-QUAL O NUMERO DE ESTAGIOS DA ERLANG *,/)
    READ(TERM,120) NESTAG
    WRITE(JTERM,130) NESTAG
    CALL DSTRB(VETYY(2,NDIST),ERLNG,NMED,NESTAG,0)
    GO TO 6200
6400 WRITE(JTERM,6410)
6410 FORMAT(1H , *-QUAL O LIMITE INFERIOR DE VARIACAO DA UNIFORME *,/)
    READ(TERM,120) LINF
    WRITE(JTERM,130) LINF
    WRITE(JTERM,6420)
6420 FORMAT(1H , *-QUAL O LIMITE SUPERIOR DE VARIACAO DA UNIFORME *,/)
    READ(TERM,120) LSUP
    WRITE(JTERM,130) LSUP
    CALL DSTRB(VETYY(2,NDIST),UNFRM,LINF,LSUP,0)
    GO TO 6200
6500 WRITE(JTERM,6510)
6510 FORMAT(1H , *-QUAL O NUMERO DE CLASSES DA EMPIRICA *,/)
6512 READ(TERM,420) NCLAS
    WRITE(JTERM,425) NCLAS
    NFRE=0
    ICONT=0
6515 ICONT=ICONT+1
    IF(ICONT.GT.NCLAS+1) GO TO 6540

```

```

WRITE(JTERM,6520)
6520 FORMAT(1H , *-QUAL O VALOR AMOSTRADO (O PARA PARAR) *,/)
READ(TERM,120) IVAL
WRITE(JTERM,130) IVAL
IF(IVAL.EQ.ZERO) GO TO 6530
VALOR(ICONT)=IVAL
GO TO 6515
6525 NFRE=NFRE+1
IF(NFRE.GT.NCLAS+1) GO TO 6540
6530 WRITE(JTERM,910)
READ(TERM,420) IFREQ
WRITE(JTERM,425) IFREQ
IF(IFREQ.EQ.0) GO TO 6560
FREQC(NFRE)=IFREQ
GO TO 6525
6540 WRITE(6550)
6550 FORMAT(1H , *-O NUMERO DE DADOS EXCEDE O NUMERO DE CLASSES. *,/,1
*H , * ENTRE COM O NUMERO DE CLASSES E OS DADOS NOVAMENTE. *,/)
GO TO 6512
6560 CALL DSTRB(VETYY(2,NDIST),EMPCA,NCLAS,0,0)
GO TO 6200

```

C

```

7000 CONTINUE
WRITE(JTERM,7100)
7100 FORMAT(1H , *-QUE PARAMETRO DESEJA ANOTAR (MEDIA, MAXIMO, MINIMO,
*,/,1H * DESVIO PADRAO, TAMANHO OU NENHUM) *,/)
7105 READ(TERM,1020) KFLAG
WRITE(JTERM,1030) KFLAG
IF(KFLAG(1).EQ.IDENT(17)) GO TO 7111
IF(KFLAG(1).EQ.IDENT(18)) GO TO 7112
IF(KFLAG(1).EQ.IDENT(19)) GO TO 7113
IF(KFLAG(1).EQ.IDENT(20)) GO TO 7114
IF(KFLAG(1).EQ.IDENT(21)) GO TO 7115
IF(KFLAG(1).EQ.IDENT(10)) GO TO 5
WRITE(JTERM,1040) KFLAG
GO TO 7105
7111 IT=1
GO TO 7116
7112 IT=2
GO TO 7116
7113 IT=3
GO TO 7116
7114 IT=4
GO TO 7116
7115 IT=5
7116 WRITE(JTERM,410)
READ(TERM,420) NEST
WRITE(JTERM,425) NEST
WRITE(JTERM,440)
READ(TERM,420) NH
WRITE(JTERM,425) NH
WRITE(JTERM,7300)
7300 FORMAT(1H , *-EM QUAL DAS SERIES SERA ANOTADO *,/)
READ(TERM,220) NS
WRITE(JTERM,225) NS
IF(IT.EQ.1) KVAL=MEDIA(VETYY(1,NEST))
IF(IT.EQ.2) KVAL=MAIOR(VETYY(1,NEST))
IF(IT.EQ.3) KVAL=MENOR(VETYY(1,NEST))
IF(IT.EQ.4) KVAL=DSVIO(VETYY(1,NEST))

```

```

IF(IT.EQ.5) KVAL=NAMST(VETYY(1,NEST))
CALL ANOTE(KVAL,NS,VETYY(4,NH))
GO TO 7000

```

```

C
8000 CONTINUE
WRITE(JTERM,8100)
8100 FORMAT(1H , *-QUANTAS SERIES DE VALORES TERA O HISTORICO *,/)
READ(TERM,220) NS
WRITE(JTERM,225) NS
WRITE(JTERM,8200)
8200 FORMAT(1H , *-QUAL O NUMERO MAXIMO DE VALORES A SEREM ANOTADOS *,/)
READ(TERM,120) NV
WRITE(JTERM,130) NV
CALL HSTRC(IHIST,NS,NV)
GO TO 5

```

```

C
9000 CONTINUE
WRITE(JTERM,9100)
9100 FORMAT(1H , *-E DISTRIBUICAO, SEQUENCIA ALEATORIA OU NENHUMA DELAS
* *,/)
9200 READ(TERM,20) IFLAG
WRITE(JTERM,25) IFLAG
IF(IFLAG(1).EQ.IDENT(12)) GO TO 9300
IF(IFLAG(1).EQ.IDENT(22)) GO TO 9400
IF(IFLAG(1).EQ.IDENT(10)) GO TO 5
WRITE(JTERM,40) (IFLAG(I),I=1,4)
GO TO 9200
9300 KT=2
GO TO 9500
9400 KT=10
9500 WRITE(JTERM,9600)
9600 FORMAT(1H , *-QUAL E A ORDEM DE DEFINICAO DA MESMA *,/)
READ(TERM,420) INUM
WRITE(JTERM,425) INUM
WRITE(JTERM,9700)
9700 FORMAT(1H , *-QUAL O NOVO VALOR DA SEMENTE *,/)
READ(TERM,120) ISEM
WRITE(JTERM,130) ISEM
CALL MUDE(VETYY(KT,INUM),ISEM)
GO TO 9000

```

```

C
10000 CONTINUE
WRITE(JTERM,9100)
10100 READ(TERM,20) IFLAG
WRITE(JTERM,25) IFLAG
IF(IFLAG(1).EQ.IDENT(12)) GO TO 10200
IF(IFLAG(1).EQ.IDENT(22)) GO TO 10300
IF(IFLAG(1).EQ.IDENT(10)) GO TO 5
WRITE(JTERM,40) (IFLAG(I),I=1,4)
GO TO 10100
10200 KT=2
GO TO 10400
10300 KT=10
10400 WRITE(JTERM,9600)
READ(TERM,420) INUM
WRITE(JTERM,425) INUM
CALL RENCZ(VETYY(KT,INUM))
GO TO 10000

```

C

```

11000 CONTINUE
      WRITE(JTERM,9100)
11100 READ(TERM,20) IFLAG
      WRITE(JTERM,25) IFLAG
      IF(IFLAG(1).EQ.IDENT(12)) GO TO 11200
      IF(IFLAG(1).EQ.IDENT(22)) GO TO 11300
      IF(IFLAG(1).EQ.IDENT(10)) GO TO 5
      WRITE(JTERM,40) (IFLAG(I),I=1,4)
      GO TO 11100
11200 KT=2
      GO TO 11400
11300 KT=10
11400 WRITE(JTERM,9600)
      READ(TERM,420) INUM
      WRITE(JTERM,425) INUM
      CALL ANTET(VETYY(KT,INUM))
      GO TO 11000
C
12000 CONTINUE
      WRITE(JTERM,9100)
12100 READ(TERM,20) IFLAG
      WRITE(JTERM,25) IFLAG
      IF(IFLAG(1).EQ.IDENT(12)) GO TO 12200
      IF(IFLAG(1).EQ.IDENT(22)) GO TO 12300
      IF(IFLAG(1).EQ.IDENT(10)) GO TO 5
      WRITE(JTERM,40) (IFLAG(I),I=1,4)
      GO TO 12100
12200 KT=2
      GO TO 12400
12300 KT=10
12400 WRITE(JTERM,9600)
      READ(TERM,420) INUM
      WRITE(JTERM,425) INUM
      CALL NATET(VETYY(KT,INUM))
      GO TO 12000
C
13000 CONTINUE
      WRITE(JTERM,13010)
13010 FORMAT(1H , '-O ELEMENTO PROCURADO PERTENCE A UM GRUPO OU CONJUNTO'
*, /)
13020 READ(TERM,2020) INOME
      WRITE(JTERM,2030) INOME
      IF(INOME(1).EQ.IDENT(25)) GO TO 13030
      IF(INOME(1).EQ.IDENT(26)) GO TO 13180
      WRITE(JTERM,2040) INOME
      GO TO 13020
13030 JJ=010
13035 WRITE(JTERM,13040)
13040 FORMAT(1H , '-QUAL A ORDEM DE DEFINICAO DO GRUPO (0 PARA PARAR)', /
*)
      READ(TERM,420) NORDEF
      WRITE(JTERM,425) NORDEF
      IF(NORDEF.EQ.ZERO) GO TO 5
      WRITE(JTERM,13050)
13050 FORMAT(1H , '-O ATRIBUTO PESQUISADO E TAMANHO OU OUTRO QUALQUER ', /
*)
13060 READ(TERM,20) IFLAG
      WRITE(JTERM,25) IFLAG
      IF(IFLAG(1).EQ.IDENT(21)) GO TO 13070

```

```

IF(IFLAG(1).EQ.IDENT(27)) GO TO 13080
WRITE(JTERM,40) IFLAG
GO TO 13060
13070 NATRIB=-3
GO TO 13095
13080 WRITE(JTERM,13090)
13090 FORMAT(1H , *-QUAL DELES (DE A ORDEM DE DEFINICAO DO MESMO)*,/)
READ(TERM,420) NATRIB
WRITE(JTERM,425) NATRIB
13095 WRITE(JTERM,13100)
13100 FORMAT(1H , *-QUAL O VALOR A SER COMPARADO (MAXIMO,MINIMO OU OUTRO
* QUALQUER)*,/)
13105 READ(TERM,20) IFLAG
WRITE(JTERM,25) IFLAG
IF(IFLAG(1).EQ.IDENT(18)) GO TO 13110
IF(IFLAG(1).EQ.IDENT(19)) GO TO 13120
IF(IFLAG(1).EQ.IDENT(27)) GO TO 13130
WRITE(JTERM,40) IFLAG
GO TO 13105
13110 NVALOR=INF TO
GO TO 13140
13120 NVALOR=-INF TO
GO TO 13140
13130 WRITE(JTERM,1100)
READ(TERM,120) NVALOR
WRITE(JTERM,130) NVALOR
13140 CALL PCURE(VETYY(JJ,NORDEF),NATRIB,NVALOR,NUMERO)
IF(NUMERO.EQ.ZERO) GO TO 13200
IF(JJ.EQ.SEIS) GO TO 13145
KATUCA=IABS(VETYY(OITO,NORDEF+25))
IF(KATUCA.EQ.UM) KK=SEIS
IF(KATUCA.EQ.DDIS) KK=CINCO
IF(KATUCA.EQ.TRES) KK=SETE
IF(KATUCA.EQ.QUATR) KK=NOVE
IF(KATUCA.LE.ZERO.OR.KATUCA.GE.CINCO) CALL ERRO(1695)
13145 CONTINUE
IF(JJ.EQ.SEIS) KTTO=50
IF(JJ.EQ.OITO) KTTO=25
IF(JJ.EQ.SEIS) KK=SETE
DO 13150 I=1,KTTO
INDICA=I
IF(VETYY(KK,I).EQ.NUMERO) GO TO 13160
13150 CONTINUE
CALL ERRO(1695)
13160 WRITE(JTERM,13170) INDICA
13170 FORMAT(1H , *-O ELEMENTO FOI ENCONTRADO. SUA ORDEM DE DEFINICAO E *
*,I2,/)
IF(JJ.EQ.SEIS) GO TO 13185
IF(JJ.EQ.OITO) GO TO 13035
CALL ERRO(1695)
13180 JJ=SEIS
13185 WRITE(JTERM,2005)
READ(TERM,420) NORDEF
WRITE(JTERM,425) NORDEF
IF(NORDEF.EQ.ZERO) GO TO 5
WRITE(JTERM,13190)
13190 FORMAT(1H , *-QUAL ATRIBUTO (DE A ORDEM DE DEFINICAO DO MESMO)*,/)
READ(TERM,420) NATRIB
WRITE(JTERM,425) NATRIB

```

```

GO TO 13095
13200 WRITE(JTERM,13210)
13210 FORMAT(1H , *-NAO FOI ENCONTRADO NENHUM ELEMENTO COM AS CARACTERIST
*ICAS DADAS. *,/,1H , *-RE-EXAMINE AS ETAPAS PRECEDENTES E VOLTE A TE
*NTAR. *,/)
GO TO 5
C
14000 CONTINUE
WRITE(JTERM,2110)
14010 READ(TERM,1020) KFLAG
WRITE(JTERM,1030) KFLAG
IF(KFLAG(1).EQ.IDENT(1)) GO TO 14020
IF(KFLAG(1).EQ.IDENT(2)) GO TO 14030
WRITE(JTERM,1040) KFLAG
GO TO 14010
14020 INDICE=CINCO
GO TO 14040
14030 INDICE=SETE
14040 WRITE(JTERM,14050)
14050 FORMAT(1H , *-QUAL A ORDEM DE DEFINICAO DESTE ELEMENTO *,/)
READ(TERM,420) IORDEM
WRITE(JTERM,425) IORDEM
14055 WRITE(JTERM,1070)
READ(TERM,420) NATRIB
WRITE(JTERM,425) NATRIB
IF(NATRIB.EQ.ZERO) GO TO 5
IVALOR=0(NATRIB,VETYY(INDICE, IORDEM))
WRITE(JTERM,14060) NATRIB, IORDEM, KFLAG, IVALOR
14060 FORMAT(1H , *-0 *,I2, * ATRIBUTO DA *,I2,1X,3A4, * CRIADA OU DEFINIDA
* E = *,I6,/)
GO TO 14055
C
15000 CONTINUE
15005 WRITE(JTERM,2005)
READ(TERM,420) NCONJ
WRITE(JTERM,425) NCONJ
IF(NCONJ.EQ.ZERO) GO TO 5
WRITE(JTERM,15010)
15010 FORMAT(1H , *-QUAL O TIPO DE ELEMENTO A SER PERMUTADO *,/,1H , *- (EN
*TIPODE, TRANSACAO, CONJUNTO, PONTEIRO OU GRUPO) *,/)
15020 READ(TERM,1020) KFLAG
WRITE(JTERM,1030) KFLAG
IF(KFLAG(1).EQ.IDENT(1)) GO TO 15030
IF(KFLAG(1).EQ.IDENT(2)) GO TO 15040
IF(KFLAG(1).EQ.IDENT(26)) GO TO 15050
IF(KFLAG(1).EQ.IDENT(34)) GO TO 15060
IF(KFLAG(1).EQ.IDENT(25)) GO TO 15070
WRITE(JTERM,1040) KFLAG
GO TO 15020
15030 NT=CINCO
GO TO 15080
15040 NT=SETE
GO TO 15080
15050 NT=SEIS
GO TO 15080
15060 NT=NOVE
GO TO 15080
15070 NT=OITO
15080 WRITE(JTERM,14050)

```

```

READ(TERM,420) IORDEM
WRITE(JTERM,425) IORDEM
WRITE(JTERM,15090)
15090 FORMAT(IH, '-EM QUE ORDEM (CRESCENTE OU DECRESCENTE)',/)
15100 READ(TERM,1020) KFLAG
WRITE(JTERM,1030) KFLAG
IF(KFLAG(1).EQ.IDENT(35)) GO TO 15110
IF(KFLAG(1).EQ.IDENT(36)) GO TO 15120
WRITE(JTERM,1040) KFLAG
GO TO 15100
15110 KT=UM
GO TO 15130
15120 KT=DOIS
15130 WRITE(JTERM,15140)
15140 FORMAT(IH, '-QUAL ATRIBUTO (TAMANHO PODE SER USADO CASO O ELEMENT
*O A SER PERMUTADO',/,IH ' SEJA CONJUNTO OU GRUPO. NESTE CASO DIGIT
*E -3',/)
READ(TERM,420) NATRIB
WRITE(JTERM,425) NATRIB
CALL PSCNE(VETYY(NT, IORDEM), VETYY(SEIS, NCONJ), KT, NATRIB)
GO TO 15005
C
16000 CONTINUE
16005 WRITE(JTERM,2005)
READ(TERM,420) NCONJ
WRITE(JTERM,425) NCONJ
IF(NCONJ.EQ.ZERO) GO TO 5
WRITE(JTERM,15090)
16010 READ(TERM,1020) KFLAG
WRITE(JTERM,1030) KFLAG
IF(KFLAG(1).EQ.IDENT(35)) GO TO 16020
IF(KFLAG(1).EQ.IDENT(36)) GO TO 16030
WRITE(JTERM,1040) KFLAG
GO TO 16010
16020 KT=UM
GO TO 16040
16030 KT=DOIS
16040 WRITE(JTERM,16050)
16050 FORMAT(IH, '-QUAL ATRIBUTO (TAMANHO PODE SER USADO. NESTE CASO DI
*GITE -3',/)
READ(TERM,420) NATRIB
WRITE(JTERM,425) NATRIB
CALL ORDNE(VETYY(SEIS, NCONJ), KT, NATRIB)
GO TO 16005
C
17000 CONTINUE
17005 WRITE(JTERM,1120)
READ(TERM,420) NT
WRITE(JTERM,425) NT
IF(NT.EQ.ZERO) GO TO 5
KSALVA=VETYY(SETE, NT)
CALL DSTRU(KSALVA)
GO TO 17005
C
END

```