

UNIVERSIDADE FEDERAL DE ITAJUBÁ – UNIFEI
INSTITUTO DE CIÊNCIAS EXATAS – ICE
DEPARTAMENTO DE MATEMÁTICA E COMPUTAÇÃO – DMC
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

MANUAL DO SISTEMA OPERACIONAL SOMAT

Autores Hipólito Douglas França Moreira e Wanessa da Silva

Orientador Cláudio Kirner

2010

ÍNDICE

1. O que é o SOMAT	2
2. Instalação do SOMAT	2
3. Interface do SOMAT	3
3.1. Visão do Usuário	3
3.2. Visão do Supervisor	4
3.3. Visão do Sistema Operacional	4
3.4. Visão do Hardware	4
4. Computador Simplificado	5
4.1. Estrutura do Hardware	6
4.2. Repertório de Instruções	9
5. Programas do Usuário	11
5.1. Linguagem de Controle de Programa	11
5.2. Programa de Usuário (nível do processo de leitura)	13
5.3. Programa de Usuário (no disco)	13
5.4. Programa de Usuário (na memória)	13
5.5. Execução do Programa Usuário (área de código, área de dados, área de impressão, leitura de dados, impressão, uso de rascunho)	14
5.6. Exemplo de um programa	15
6. Editor do SOMAT	17
6.1. Interface	17
6.2. Inserção de um Programa	17
6.2.1. Por edição	17
6.2.2. Começando do zero	18
6.3. Inserção de um lote	18
7. Execução de Programas no SOMAT	19
7.1. Execução de um programa	19
7.2. Execução de um lote	20
7.3. Visualização durante a execução.....	20
8. Visualização de um Programa Executado	21
9. Discussão do Uso do SOMAT para a Aprendizagem de Sistemas Operacionais	22

1. O QUE É O SOMAT

O Sistema Operacional Multiprogramado Aberto para Treinamento (SOMAT) é um sistema operacional multiprogramado com execução de programas de usuário em tempo compartilhado que opera através de compartilhamento da UCP entre seus diversos processos concorrentes e os programas de usuários provenientes de terminais.

O sistema operacional SOMAT está montado sobre um hardware simulado, contendo sete processos concorrentes, que atuam sobre periféricos, buffers, disco, memória, e outros recursos, como blocos de controle de programas, variáveis de controle etc. Os programas do usuário são montados em Assembly.

2. INSTALAÇÃO DO SOMAT

O SOMAT originalmente criado para executar no sistema DOS, deve ser adaptado para ser executado nos computadores atuais, devido ao aumento considerável de tecnologia, pois a capacidade de processamento atual acelera o SOMAT e não permite seu uso.

Para executar o SOMAT em velocidade de processamento adequado, deve-se baixar o software de simulação de sistema DOS, o DOSBox, disponível em <http://www.dosbox.com/download.php?main=1>.

O SOMAT se encontra disponível em www.ckirner.com/apoio/so/somat.zip.

Para executar o SOMAT no DOSBox no sistema operacional windows, é necessário montar um drive contendo no DOSBox contendo o caminho do SOMAT, pelo seguinte código:

```
Z:\>mount c c:\users\...\somat (coloque o caminho do somat aqui)
```

```
Z:\>c:
```

```
C:\>somat
```

Observação: um caminho mais rápido e prático para executar o SOMAT, apenas coloque um atalho do aplicativo DOSBox na pasta do SOMAT e quando for abrir o aplicativo arraste o arquivo somat.exe sobre o atalho do DOSBox.

Para executar o DOSBox no sistema operacional Linux, apenas clique com o botão direito do mouse sobre o somat.exe e nas opções escolha <abrir com outras aplicações> e

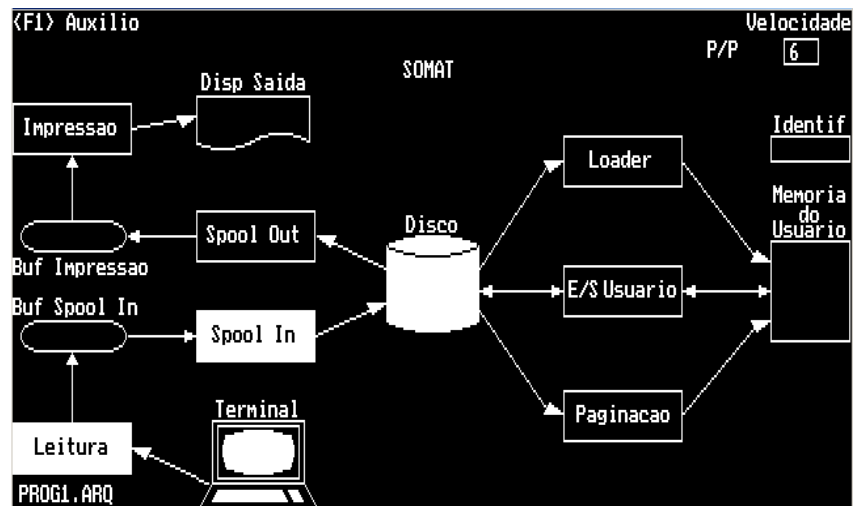
3.2 Visão do Supervisor

A visão do supervisor mostra as principais tarefas realizadas pelo SOMAT, podendo o super-usuário interferir nestas.



3.3 Visão do Sistema Operacional

Mostra a transição e o funcionamento de todos os componentes do sistema operacional



3.4 Visão do Hardware

Aqui é mostrado como o hardware mantém o funcionamento do SOMAT. É mostrado a memória física a tabela de páginas e os registradores do processador e a sua interação entre eles.

4.1. Estrutura do Hardware

O hardware do computador simplificado implementado por software através de procedimentos e estruturas de dados, simula um computador, contendo os recursos necessários para suportar multiprogramação, além de ter um sistema de segurança de dados, onde ocorre o isolamento da parte de memória responsável por cada job, não deixando que um leia ou acesse informações de outros que estão rodando simultaneamente a este.

Seus componentes de hardware incluem:

- uma unidade central de processamento (CPU);
- memória principal de acesso rápido;
- uma unidade de armazenamento externo (disco);
- contador do tempo para execução de programas (timer);
- contador da fatia de tempo da CPU para a execução de cada programa (time slice);
- relógio (clock);
- unidade de entrada e saída de dados controladas por canais autônomos;
- vetor de interrupções;

O Hardware simulado restringe os detalhes da implementação do SO somente as suas características funcionais e abstrai detalhes de baixo nível de sua construção.

Desta forma, a unidade básica do sistema de armazenamento (palavra), é especificada através de uma estrutura de dados.

Uma palavra do CS é um registro contendo três campos para valores inteiros:

Palavra: | C1 | C2 | C3 |

Ao armazenar uma palavra de instrução, os campos de uma palavra terão os seguintes significados:

C1: código da operação

C2: página referida pela instrução

C3: deslocamento dentro da página referida (palavra)

Quando representar um dado, somente o primeiro campo da palavra será relevante, armazenando o valor do dado:

C1: valor do dado armazenado

C2: valor do dado armazenado (irrelevante)

C3: valor do dado armazenado (irrelevante)

Para facilidade de desenvolvimento do sistema simulado, foi definida uma memória do usuário, composta por 256 palavras e sujeita a controles e manipulação por parte do sistema operacional. Para a implementação de um sistema de paginação e memória virtual, entretanto o sistema operacional irá estruturá-la em blocos de 8 palavras, formando 32 páginas.

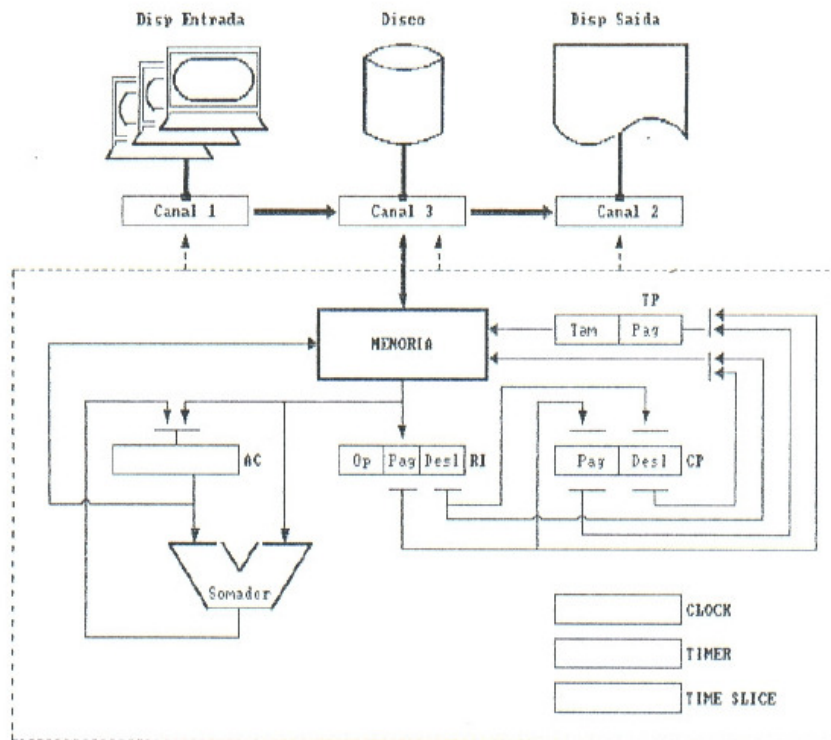
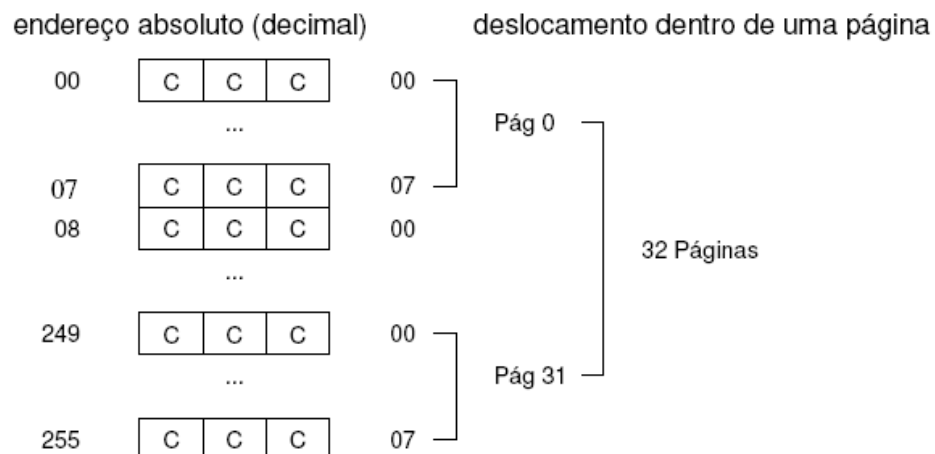


Figura – Diagrama do fluxo de dados no hardware simulado.

Desta forma, um endereço qualquer valido XYZ se refere à palavra Z, dentro da página.



Com a finalidade de permitir que mais programas possam ser posicionados na memória simultaneamente para execução, o Sistema Operacional implementa no sistema de endereçamento relativo, com memória compartilhada através de paginação.

Esse sistema de paginação faz com que os programas sejam tratados por partes, de modo que seus códigos não necessitam estar completamente carregados em memória durante as suas execuções e nem estejam alocados de maneira seqüencial, ocupando espaços contíguos na memória.

Os programas são divididos em páginas, do mesmo tamanho que as páginas de memória, que são carregadas em espaços disponíveis da memória ou sobrepondo outros pedaços de programa, a medida em que são necessárias para a continuidade da execução dos programas.

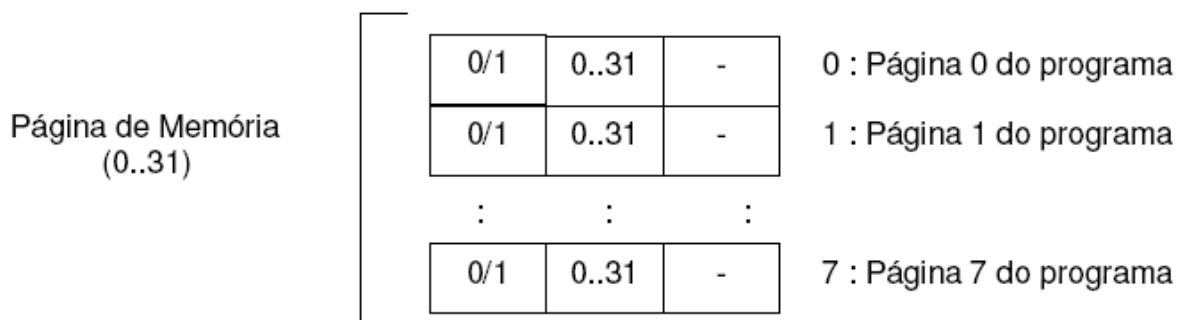
Para poder identificar a presença das páginas dos programas na memória e a sua eventual localização, o Sistema Operacional monta então uma tabela de páginas, para cada programa, instalada em uma página comum de memória.

C1: indica presença ou ausência de uma página do programa na memória

C2: endereço efetivo da página na memória (se estiver presente)

C3: (não utilizado)

Em qualquer condição de interrupção, se houver um programa em execução, ele é interrompido logo após a realização da instrução que estava sendo tratada, e o controle é desviado para a rotina de tratamento de interrupções do sistema operacional. As informações referentes à situação dos registradores do hardware e do estado de execução do programa são salvos no seu BCP e podem ser restauradas quando ele voltar a ser executado.



Desta forma, para obter o endereço efetivo de onde se encontra uma palavra de

instrução ou de dados, de um programa o sistema operacional segue o seguinte procedimento:

- Identifica a página de memória que contém a Tabela de Páginas do programa.
- Pesquisa nessa página, na palavra correspondente à página buscada, a presença e o endereço efetivo da página.

O sistema operacional mantém uma Tabela de Páginas para cada programa em execução. Por ser instalada em uma página comum de memória, entretanto, o tamanho dos programas é limitado a, no Máximo, 8 páginas.

A Unidade Central de Processamento do CS foi projetada com o intuito de prover suporte de Hardware necessário a execução de vários programas com tempo compartilhado e paginação de memória do usuário.

4.2. Repertório de Instruções

A Unidade Central de processamento do SOMAT opera com um conjunto de instruções simplificadas que dão suporte ao programa, do usuário, que envolve operações aritméticas, controle de desvio e parada de programas, manipulação de posições de memória, e operações de E/S e ainda algumas das instruções permitem o endereçamento de páginas de memória.

No SOMAT são implementados os seguintes repertórios de instruções:

HLT: Indica o fim de processamento de um programa.

RD XY 0: Lê a página corrente a área de dados do programa no disco e carrega seu conteúdo na página XY do programa

PRN XY 0: Imprime, na área de impressão do disco alocada pelo programa, o conteúdo da página XY desse programa.

LD XYZ: Carrega no acumulador (ACC) o conteúdo da palavra Z da página XY do programa sendo executado. No caso do sistema simulado, a operação carrega no acumulador somente o conteúdo do primeiro campo da palavra de dados.

STR XYZ: Armazena o conteúdo do acumulador na palavra Z da página XY do programa. Atribui o valor do primeiro campo da palavra.

SUB XYZ: Atribui ao acumulador o valor correspondente ao valor que continha, subtraído do valor armazenado no primeiro campo da palavra Z da página programa do usuário.

ADD XYZ: Idem à subtração, atribuindo ao acumulador o resultado da soma ao invés da subtração.

JMP XYZ: Atribui um novo valor para o Contador de Programa do programa em execução, que passa a ser executado a partir dessa posição.

JNG XYZ: O conteúdo do acumulador é checado e, caso contenha um valor negativo, o valor de XYZ é atribuído ao CP. Caso o acumulador não seja negativo, o programa continua sua execução a partir da próxima instrução em relação ao endereço em que encontra.

O SOMAT contém ainda um registrador de interrupções que indicam anomalias ou sinais de alerta gerados pelo hardware em determinadas condições.

O registrador de interrupções é implementado através de um vetor de 12 posições com campos que podem assumir os valores 0 ou 1. O valor 1 em um elemento desse vetor representa a ocorrência da interrupção a que corresponde. O valor 0 indica a não ocorrência de uma interrupção.

Vetor de Interrupções		
0	0/1	Proteção de Memória
1	0/1	Código de operação inválido
2	0/1	Overflow
3	0/1	Falta de página
4	0/1	Timer
5	0/1	TS
6	0/1	Leia
7	0/1	Imprima
8	0/1	Pare
9	0/1	Canal 1 (Leitora)
10	0/1	Canal 2 (Impressora)
11	0/1	Canal 3 (Disco)
12	0/1	Interrupção Externa

Os elementos de 0 a 3 do vetor correspondem a interrupções de programa, que são geradas em decorrência das suas execuções.

O valor 1 no elemento 0 do vetor indica que o programa tentou acessar uma posição de memória fora da área do seu programa, ou tentou ler ou imprimir dados do disco numa área fora do espaço que havia alocado.

O elemento 1 do vetor recebe o valor 1 sempre que um programa tentar executar uma operação cujo código não é reconhecido pelo sistema operacional.

Se, durante a operação de uma instrução aritmética, houver overflow no valor armazenado no acumulador, o elemento 2 do vetor de interrupções receberá o valor 1.

Caso, no cálculo do endereço efetivo de uma posição de memória referenciada por um programa, sua tabela de páginas indique que a página não está presente na memória, uma interrupção por falta de página é gerada, atribuindo-se o valor 1 ao elemento 3 do vetor.

TIMER e TS são interrupções de tempo e são geradas quando esses registradores assumem o valor 0.

Leia, Imprima e Pare são interrupções do sistema e indicam a necessidade de realizações de ações do sistema operacional.

As interrupções dos canais de controle dos periféricos indicam ao sistema operacional o estado da Leitora, da Impressora e do Disco, e são atividades sempre que estes dispositivos completam a realização de uma operação de Entrada/Saída.

Interrupções externas, geradas e são identificadas pelo usuário dos sistema, podem ser geradas e são identificadas pelo elemento 12 do vetor de interrupções.

Em qualquer condição de interrupção, se houver um programa em execução, ele é interrompido logo após a realização da instrução que estava sendo tratada, e o controle é desviado para a rotina de tratamento de interrupções do sistema operacional. As informações referentes à situação dos registradores do hardware e do estado de execução do programa são salvos no seu BCP e podem ser restauradas quando ele voltar a ser executado.

As interrupções por proteção de memória, código de operação inválido, overflow e timer são consideradas fatais e sua ocorrência durante a execução de um programa causa o término forçado de sua execução e a sua saída do sistema.

5. PROGRAMAS DO USUÁRIO

5.1. Linguagem de Controle de Programa

A linguagem de controle de programa do SOMAT define as páginas de início de job, página de início de páginas de código, página de início de páginas de dados e página de fim de job.

A página de início de job apresenta a seguinte estrutura:

Edicao						
Pagina de inicio de job						
	Tipo	Op	C1	Pag	Desl	
Pal 0 :	*	-	0	-	-	: Job
Pal 1 :	*	-	nJob	-	-	: Nro Job
Pal 2 :	*	-	T	-	-	: Timer
Pal 3 :	*	-	Tam	-	-	: NroPagCod
Pal 4 :	*	-	Rasc	-	-	: NroPagRasc
Pal 5 :	*	-	Impr	-	-	: NroPag Impr
Pal 6 :	-	-	-	-	-	: -
Pal 7 :	-	-	-	-	-	: -
Esc ↑↓						

Na palavra 0 o uso de 0, marca a página utilizada como página de job;

Na palavra 1 está a identificação do job no sistema;

Na palavra 2 está o timer, que define o tempo necessário para execução do job;

Na palavra 3 está o número de páginas de código, a qual não deve estar diferente da quantidade de páginas de código, ou então o sistema interromperá e finalizará o job;

Na palavra 4 está o número de páginas de rascunho, a qual devem ser incluídas as páginas de dados e as páginas de impressão;

Na palavra 5 está o número de páginas de impressão, as quais são reservadas para dar saída aos dados produzidos pelos programas.

A página de início de página de códigos e página de dados são definidas a seguir:

Edicao						
Pagina de inicio de codigo						
Pal 0 :	*	-	1	-	-	: Prog
Pal 1 :	-	-	-	-	-	: -
:	:	:	:	:	:	:
Pal 7 :	-	-	-	-	-	: -
Pagina de inicio de dados						
Pal 0 :	*	-	2	-	-	: Dados
Pal 1 :	-	-	-	-	-	: -
:	:	:	:	:	:	:
Esc ↑↓						
Pagina de fim de programa						
Pal 0 :	*	-	3	-	-	: Fim
Pal 1 :	-	-	-	-	-	: -
:	:	:	:	:	:	:
Pal 7 :	-	-	-	-	-	: -

Na palavra 0 o uso de 1 define a página de controle como página de início de páginas

de códigos, enquanto o uso de 2 define a página de controle como página de início de páginas de dados e o uso de 3 define a página de controle como página de fim de job.

5.2. Programa de Usuário (nível do processo de leitura)

Durante esta fase o programa de usuário é carregado pela leitora sendo lido pelo sistema a partir das identificações da linguagem de controle, reconhecendo as páginas de controle, páginas de códigos e páginas de dados.

*0 JOB
*1 CODIGO
Código
Código
*2 DADOS
Dados
Dados
*3 FIM

Em seguida o Spool in desmonta as identificações de controle e monta a estrutura de códigos e dados que serão carregados no disco.

Código
Código
Dados
Dados

5.3. Programa de Usuário (no disco)

Durante esta fase o sistema armazena os códigos e os dados vindos do spool in do processo de leitura e aloca espaço para impressão, definido na página de controle, onde as páginas serão montadas de acordo com a ordem proposta no programa.

Código	Pag 0
Código	Pag 1
Dados	Pag 0
Dados	Pag 1
Impressão	Pag 0
Impressão	Pag 1

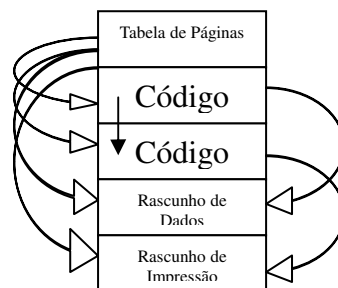
5.4. Programa de Usuário (na memória)

Quando o programa passa do disco para a memória, o sistema constrói uma tabela de

páginas, que possui a referência para todas as páginas utilizadas no job, definindo quais as páginas de dados, serão rascunho e quais serão utilizadas para impressão.

Esta definição faz com que o sistema seja organizado para evitar acessos indevidos a memória, pois uma tentativa do programa para acessar o disco acima do que foi definido ou imprimir numa página de rascunho não declarado, incorrem em interrupções do programa para proteção do sistema.

Na memória o programa do usuário apresenta um esquema onde as páginas de código e páginas de rascunho são montadas na memória do usuário.



5.5. Execução do Programa Usuário (área de código, área de dados, área de impressão, leitura de dados, impressão, uso de rascunho)

A execução do programa após ser carregado na memória, ocorre por meio da leitura das páginas de códigos e de dados e criação de páginas de impressão para a montagem de uma tabela de páginas onde são guardados os endereços das páginas.

Durante a execução dos códigos os acessos ao disco são descritos e carregam dados para as páginas de dados, porém esses acessos são limitados pelo número de chamadas declaradas anteriormente, se mais acessos a memória ocorrerem durante a execução dos códigos por meio de novas chamadas por jumps, o sistema operacional interromperá o programa.

Após carregar os dados do disco para as páginas de rascunho o código trabalha com os dados e os armazena nas páginas de impressão, que são páginas de rascunhos preparadas para dar saída, que são armazenadas de novo no disco para então serem carregadas pelo Spool Out e serem impressas.

5.6. Exemplo de um programa

Pagina 1 Tipo : *					
Palavra 0	Op :	Campo1 :	0	Pag :	0
Palavra 1	Op :	Campo1 :	1	Pag :	0
Palavra 2	Op :	Campo1 :	250	Pag :	0
Palavra 3	Op :	Campo1 :	2	Pag :	0
Palavra 4	Op :	Campo1 :	3	Pag :	0
Palavra 5	Op :	Campo1 :	1	Pag :	0
Palavra 6	Op :	Campo1 :	0	Pag :	0
Palavra 7	Op :	Campo1 :	0	Pag :	0

Página de Controle

Pagina 2 Tipo : *_					
Palavra 0	Op :	Campo1 :	1	Pag :	0
Palavra 1	Op :	Campo1 :	0	Pag :	0
Palavra 2	Op :	Campo1 :	0	Pag :	0
Palavra 3	Op :	Campo1 :	0	Pag :	0
Palavra 4	Op :	Campo1 :	0	Pag :	0
Palavra 5	Op :	Campo1 :	0	Pag :	0
Palavra 6	Op :	Campo1 :	0	Pag :	0
Palavra 7	Op :	Campo1 :	0	Pag :	0

Página de Início de páginas de códigos

Pagina 3 Tipo : C					
Palavra 0	Op :	RD	Campo1 :	1	Pag : 0
Palavra 1	Op :	RD	Campo1 :	1	Pag : 1
Palavra 2	Op :	LD	Campo1 :	3	Pag : 0
Palavra 3	Op :	SUB	Campo1 :	5	Pag : 0
Palavra 4	Op :	STR	Campo1 :	4	Pag : 0
Palavra 5	Op :	LD	Campo1 :	3	Pag : 1
Palavra 6	Op :	ADD	Campo1 :	6	Pag : 1
Palavra 7	Op :	STR	Campo1 :	4	Pag : 1

Página de Código 0

Pagina 4 Tipo : C					
Palavra 0	Op :	LD	Campo1 :	3	Pag : 0
Palavra 1	Op :	JNG	Campo1 :	8	Pag : 1
Palavra 2	Op :	JMP	Campo1 :	7	Pag : 0
Palavra 3	Op :	LD	Campo1 :	3	Pag : 1
Palavra 4	Op :	STR	Campo1 :	4	Pag : 2
Palavra 5	Op :	PRN	Campo1 :	2	Pag : 2
Palavra 6	Op :	HLT	Campo1 :	0	Pag : 0
Palavra 7	Op :	HLT	Campo1 :	0	Pag : 0

Página de Código 1

Pagina 5 Tipo : *_					
Palavra 0	Op :	Campo1 : 2	Pag : 0	Desl : 0	
Palavra 1	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 2	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 3	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 4	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 5	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 6	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 7	Op :	Campo1 : 0	Pag : 0	Desl : 0	

Página de Início de páginas de dados

Pagina 6 Tipo : D_					
Palavra 0	Op :	Campo1 : 4	Pag : 0	Desl : 0	
Palavra 1	Op :	Campo1 : 1	Pag : 0	Desl : 0	
Palavra 2	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 3	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 4	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 5	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 6	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 7	Op :	Campo1 : 0	Pag : 0	Desl : 0	

Página de Dados 0

Pagina 7 Tipo : D_					
Palavra 0	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 1	Op :	Campo1 : 4	Pag : 0	Desl : 0	
Palavra 2	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 3	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 4	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 5	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 6	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 7	Op :	Campo1 : 0	Pag : 0	Desl : 0	

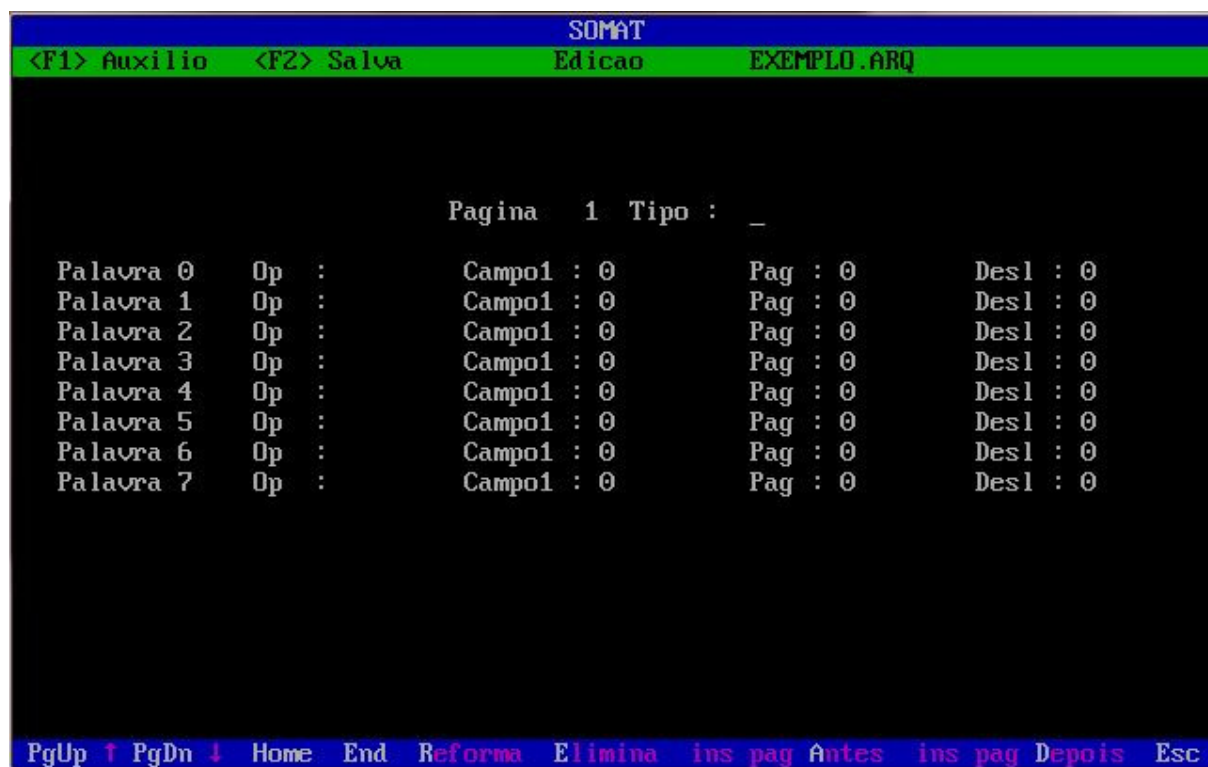
Página de Dados 1

Pagina 8 Tipo : *_					
Palavra 0	Op :	Campo1 : 3	Pag : 0	Desl : 0	
Palavra 1	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 2	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 3	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 4	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 5	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 6	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 7	Op :	Campo1 : 0	Pag : 0	Desl : 0	

Página de Fim de Programa

6. EDITOR DO SOMAT

6.1. Interface



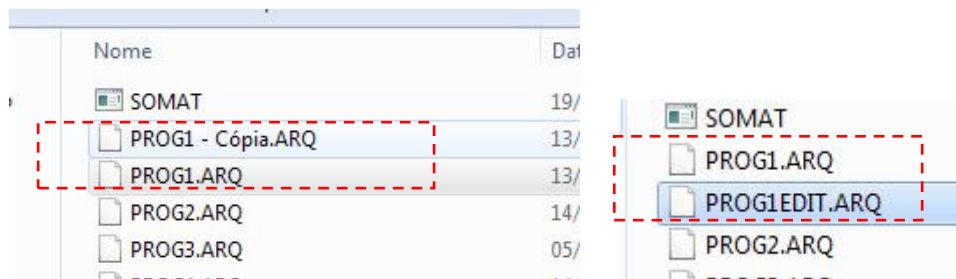
A interface do editor do SOMAT apresenta a página do programa a ser editado, mostra os comandos para ajuda (F1), salvar alterações (F2), avançar página (PageDown), recuar página (PageUp), primeira página (Home), última página (End), editar (R), eliminar página (E), inserir página antes da página atual (A), inserir página depois da página atual (D) e para sair do editor o comando ESC.

6.2. Inserção de um Programa

A inserção de um programa no SOMAT pode ser feita por meio de edição de código pré-existente e começando um programa do zero:

6.2.1 Por edição

A inserção de programas por edição é a alteração de um programa pré-existente para adição ou remoção de códigos e dados, onde o usuário copia um dos arquivos com extensão (*.ARQ) e o renomeia com o nome que desejar.



Cópia e renomeação de Arquivo ARQ

Após a criação do arquivo a ser editado ele deve ser carregado no editor do SOMAT para edição dos dados.



Edição de arquivo modificado

Depois de aberto o arquivo, a interface do editor descrita acima é apresentada ao usuário.

6.2.2 Começando do zero (opcional)

O usuário pode também iniciar seus programas do zero, onde ele não precisa copiar um programa pré-existente, ele precisa apenas na hora de chamar o editor escrever o nome do programa seguido da extensão (*.ARQ).



Criação de Novo Programa para começar do zero

Após criar o novo arquivo você é encaminhado para a interface descrita anteriormente, onde pode começar a criar seus próprios programas partindo do zero.

6.3. Inserção de um lote

Para inserir um lote a um programa já existente o usuário deve seguir os mesmos

passos para criação de um JOB, onde na página de controle deve inserir um outro identificador de JOB, por exemplo, o primeiro JOB recebeu 1, quaisquer outro JOB deve receber outro identificador que não seja o 1.

Pagina 7 Tipo : *_					
Palavra 0	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 1	Op :	Campo1 : 3	Pag : 0	Desl : 0	
Palavra 2	Op :	Campo1 : 200	Pag : 0	Desl : 0	
Palavra 3	Op :	Campo1 : 1	Pag : 0	Desl : 0	
Palavra 4	Op :	Campo1 : 1	Pag : 0	Desl : 0	
Palavra 5	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 6	Op :	Campo1 : 0	Pag : 0	Desl : 0	
Palavra 7	Op :	Campo1 : 0	Pag : 0	Desl : 0	

Inserção de uma nova página de controle

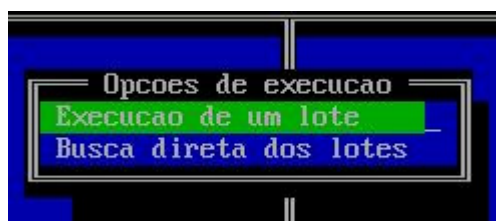
O usuário deve inserir uma página após a página de fim de programa e criar um novo programa a partir do zero, depois do programa escrito anteriormente.

7. EXECUÇÃO DE PROGRAMAS NO SOMAT

7.1. Execução de um programa

Para executar um programa o usuário deve na tela principal do programa apertar a tecla X, ou selecionar execução, onde aparecerá um caixa solicitando o processo de execução, se em lotes (execução de único arquivo), ou pela busca direta.

Para a execução de um programa selecione execução de um lote:



Depois aparecerá um campo pedindo o nome do arquivo a ser executado:

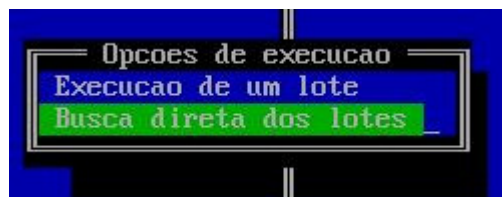


Depois de inserido o nome do arquivo, o usuário deve solicitar, ou não o uso de impressão, onde deve responder sim ou não, em caso não deve especificar um arquivo de saída, em caso sim, o sistema não mais permite o acesso à impressora.



7.2. Execução de um lote

Para execução de um lote de arquivos o usuário deve selecionar a caixa busca direta dos lotes:



Onde aparece uma caixa solicitando o diretório (pasta) onde se localizam os lotes, deixar o campo vazio significa que os arquivos estão no diretório raiz (pasta do SOMAT).

Após apertar para continuar será questionado sobre a extensão dos arquivos (usamos no SOMAT arquivos com extensão *.ARQ).



O SOMAT seguirá os mesmos passos de solicitação de impressora e arquivo de saída da execução de apenas um lote, descritos anteriormente.

7.3. Visualização durante a execução (mudanças)

Durante a execução do programa o usuário pode visualizá-lo de quatro perspectivas distintas, que durante a execução do programa podem ser alternadas.



Na visão do usuário, o usuário enxerga apenas as chamadas aos processos, vendo a carga dos seus programas e as áreas do sistema operacional que estão ativas.

Na visão do supervisor o usuário enxerga o início dos processos, os processos

necessários a execução dos programas.

Na visão do sistema operacional mostra como os dados trafegam no sistema operacional, bem como o caminho percorrido do terminal de entrada até os processos de saída dos programas.

Na visão do hardware, o usuário enxerga como os dados são manipulados e organizados na memória, sua alocação e dispositivos envolvidos no funcionamento.

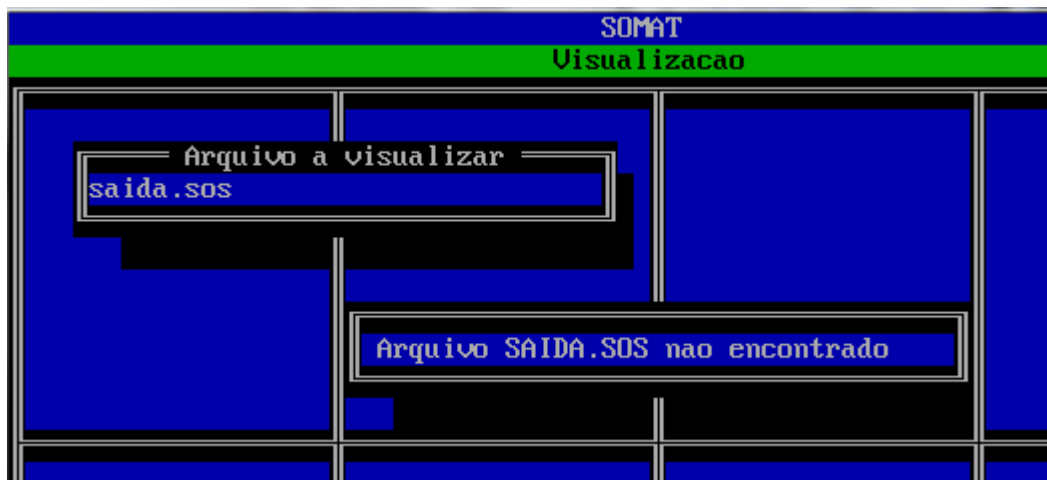
8. VISUALIZAÇÃO DE UM PROGRAMA EXECUTADO

Para visualizar um programa que executado o usuário deve ter criado um arquivo de saída durante o processo de execução, cuja extensão compreende (*.SOS), para poder visualizar os resultados do programa ou lote de programas executados.

Para visualizar o arquivo o usuário deve abrir a aba Visualização na interface principal do SOMAT, por meio da tecla “V”, ou pelas setas do teclado, onde aparecerá uma caixa solicitando o arquivo de saída criado anteriormente.



Caso não o tenha criado surgirá uma caixa avisando sobre a não existência do arquivo:



Na visualização aparecem as páginas de códigos, páginas de dados, páginas de impressão, identificador de job e tratamento de interrupção:

Job Numero : 1			Paginas de Dados	
Inicio do Programa			4	
RD	0	0	1	
LD	0	0	0	
SUB	0	1	4	
STR	0	0	0	
LD	0	2	0	
ADD	0	3	0	
STR	0	2	0	
LD	0	0		
			Paginas de Impressao	
JNG	1	2	20	
JMP	0	1	0	
LD	0	2	0	
STR	1	0	0	
PRN	1	0	0	
HLT	0	0	0	
HLT	0	0	0	
HLT	0	0	0	
			Impressao de Mensagem	
			Interrupcao Ocorrida	
			Fim de Programa	
			Job Num : 1	

Arquivo de visualização

Onde aparece a descrição dos códigos, as páginas acessadas pelos códigos e seus deslocamentos, os dados contidos nas páginas de dados, os resultados na página de impressão e exibição do tratamento da interrupção, que no caso do exemplo mostra que o programa terminou na página de fim de programa sem violar as regras do sistema operacional.

9. DISCUSSÃO DO USO DO SOMAT PARA O APRENDIZADO DE SISTEMA OPERACIONAL

O uso do SOMAT (Sistema Operacional Multiprogramado Aberto Para Treinamento) para o aprendizado de sistemas operacionais trás benefícios tanto para o professor que ministra a matéria, quanto para o aluno que a aprende. Isso porque o simulador SOMAT mostra de maneira simplificada o que acontece em um sistema operacional real, com isso o professor pode apresentar os conceitos e mecanismos de um sistema operacional real através não só de maneira teórica como também prática, o que motiva mais os alunos e gera um ambiente de aula mais dinâmico.

Com essa ferramenta visual o aluno pode, de uma maneira mais fácil, entender os conceitos teóricos abordados pelo professor, através do uso de uma ferramenta que simula um SO e trás consigo implementados alguns dos conceitos mais importantes de um Sistema Operacional real, pode também visualizar a dinâmica do Sistema Operacional de várias formas, através de vários tipos de visões, pode alterar programas já existentes e pode até fazer seu próprio programa e vê-lo rodar nessa ferramenta simplificada.

O professor encontra nessa ferramenta, o SOMAT, um apoio para conseguir passar aos alunos toda a dinâmica de um Sistema Operacional, através de um método mais interativo, diminuindo a distância entre um modelo de Sistema Operacional real e conceitual, tornando este último mais real, fazendo com que o aprendizado de Sistemas Operacionais se torne mais efetivo e interessante.

Sem dúvida o Sistema Operacional SOMAT é um grande aliado ao aprendizado de Sistema Operacional e uma grande ferramenta, a disposição dos professores, de auxílio e complemento, para os alunos, neste aprendizado.