

SISTEMA OPERACIONAL
EDIX E INFOWORD
VERSÃO 1.0



Empresa Brasileira de Pesquisa Agropecuária - EMBRAPA
Vinculada ao Ministério da Agricultura
Centro Nacional de Pesquisa de Suínos e Aves - CNPSA

1 - SC

Sistema operacional EDIX e
1988 FL-12738



14051-1

REPÚBLICA FEDERATIVA DO BRASIL

Presidente: José Sarney

Ministro da Agricultura: Iris Rezende Machado

Empresa Brasileira de Pesquisa Agropecuária - EMBRAPA

Presidente: Ormuz Freitas Rivaldo

Diretores: Ali Aldersi Saab

Derli Chaves Machado da Silva

Francisco Ferrer Bezerra

Centro Nacional de Pesquisa de Suínos e Aves - CNPSA

Chefe: Paulo Roberto Souza da Silveira

Chefe Adjunto Técnico: Hacy Pinto Barbosa

Chefe Adjunto de Apoio: Adenir José Basso



- BIBLIOTECA -



DOCUMENTOS Nº 15

ISSN: 0101-6245

Sistema operacional EDIX e Infoword: versão 1.0

Luiz Afonso de Rossó - Programador
Paulo da Silva Pinto Júnior - Operador



Empresa Brasileira de Pesquisa Agropecuária - EMBRAPA
Vinculada ao Ministério da Agricultura
Centro Nacional de Pesquisa de Suínos e Aves - CNPSA
Concórdia - SC.

Exemplares desta publicação podem ser solicitados à

EMBRAPA-CNPSA

Br 153 - km 110 - Vila Tamanduã

Caixa Postal D-3

89.700 - Concórdia - SC.

Telefone: (0499) 440122 e 440070

Telex: (0492) 271 EBPA BR

Tiragem: 200 exemplares

EMBRAPA / DIE	
Valor aquisição Cr\$	_____
C.A. de aquisição	_____
Nº N. Fiscal da U.A.	_____
Fornecedor	_____
Nº Ordem compra	_____
Origem	_____
Nº de série	_____

Rosso, Luiz Afonso de.

Sistema operacional EDIX e Inforword: versão 1.0, por Luiz Afonso de Rosso e Paulo da Siva Pinto Junior. Concórdia, SC, EMBRAPA-CNPSA, 1988.

43p. (EMBRAPA-CNPSA, Documentos, 15)

1. Computador-linguagem. I. Pinto Junior, Paulo da Silva, colab. II. Empresa Brasileira de Pesquisa Agropecuária. Centro Nacional de Pesquisa de Suínos e Aves, Concórdia, SC. III. Título. IV. Série.

CDD 001.6424

© EMBRAPA

SUMÁRIO

	Pág.
1. INTRODUÇÃO	06
2. EDIX	06
3. COMANDOS MAIS USADOS NO SISTEMA OPERACIONAL EDIX	08
mkdir	08
rmdir	08
cd	08
ls	09
cat	10
imp	11
sus	11
pro	12
can	12
pre	12
lib	12
pri	12
mos	13
rem	13
sta	13
cp	13
mv	14
ln	14
rm	15
chmod	15
chgrp	16
chown	17
file	17
tail	17
df	18
du	18
wc	18
od	19
find	19
diff	20
comm	20
split	21
tee	22
ps	22
nohup	23
who	23
pwd	23
date	23
time	23
kill	24
sleep	24

tty	24
stty	24
sort	25
passwd	26
mount e umount	26
echo	27
cal	27
banner	27
write	27
mail	28
4. INTERPRETADOR SHELL	29
set	29
test	30
read	30
shift	31
if	31
while	31
until	32
for	32
case	33
break	33
continue	33
5. ED	34
6. INFOWORD (iw)	35
7. COMANDOS DO INFOWORD	36
Comandos do menu de abertura	36
Comandos do menu principal	37
Comandos do menu de aceleração	38
Comandos do menu de blocos	39
Comandos do menu de impressão	39
Comandos do menu de tela	40
Comandos do menu de ajuda	40
8. EXERCÍCIOS	41
APÊNDICE A	44
APÊNDICE B	45
APÊNDICE C	48
9. BIBLIOGRAFIA CONSULTADA	51

SISTEMA OPERACIONAL EDIX E INFOWORD

1. INTRODUÇÃO

O sistema operacional EDIX foi desenvolvido pela EDISA - Eletrônica Digital S/A, para a série ED-600, baseado no sistema UNIX, da Bell Laboratories. Oferece essencialmente os mesmos serviços que os outros, porém possui características importantes como multiusuários, multitarefas, interativo, interpretador de comandos, que é o shell, e utilitários com funções específicas. Pode também ser usado em uma grande variedade de computadores, flexibilidade esta que o destaca quando comparado aos outros sistemas.

Um sistema operacional compreende uma série de programas responsáveis pela execução de tarefas que facilitam a interação do usuário com ele. Eles gerenciam todo o sistema (teclado, vídeo, impressora, processadores de textos, etc.).

O EDIX é fornecido com pouco mais de 200 programas utilitários, mas somente cerca de 40 deles são suficientes para uma operação básica do computador. Assim, aqui examinam-se apenas os comandos mais usados (populares).

Destaca-se também, além do sistema operacional EDIX, os editores de textos ED e INFOWORD. Sobre o ED mostram-se apenas os comandos mais usados, mas suficientes para entender o mecanismo do mesmo. Do INFOWORD, por ser mais amplo, informa-se como criar e manipular um arquivo, gravar em disco, fazer cópias e citam-se todos os comandos dos menus.

No final, encontram-se os apêndices. O apêndice A contém uma lista de símbolos e caracteres especiais; o apêndice B contém os arquivos citados nos exercícios; e o C, contém a tabela ASCII.

Esta apostila tem o propósito de auxiliar aos usuários do EDIX, em consultas rápidas e simples, sobre os comandos mais comuns do referido sistema, além dos editores de textos ED e INFOWORD, não dispensando, sempre que necessário, a utilização dos manuais ou de outras fontes de informações mais completas.

2. EDIX

Este sistema está estruturado em torno de um núcleo, e serve para administrar e controlar as operações do computador, suprimindo todas as necessidades do mesmo, tais como: operações de entrada e saída, gerenciamento de memória entre

outras.

Possue um interpretador de comandos chamado shell, que é o padrão do sistema, sendo este utilitário considerado o mais importante do EDIX. Sua função é receber e interpretar todos os comandos digitados pelo usuário.

Os utilitários ou comandos são programas do sistema operacional, cujas funções são manipulação de arquivos, diretórios e nomes de arquivos; execução de programas; informações de estado; comunicações; processamento de textos; manipulação de informações e periféricos.

Um arquivo pode ser qualquer espaço físico ocupado em uma unidade de armazenamento de dados (disquete, fita, winchester), e referenciado por um nome. O nome de um arquivo pode ter até 14 caracteres, sendo que não se pode ter dois arquivos com o mesmo nome, no mesmo diretório. Quando se quer referir a vários arquivos, pode-se usar os símbolos "*" e "?". O "*" substitue nomes e o "?" substitue caracter.

Neste sistema pode-se ter três tipos de arquivos:

a) comuns ou ordinários, que servem para armazenar informações. Podem conter programas ou dados;

b) diretórios, que contêm a lista dos nomes dos arquivos. Contêm comuns e especiais;

c) especiais, que fazem ligação de entrada e saída com o hardware. Estão quase todos no diretório /dev. Eles não possuem nenhuma informação.

Existe no EDIX, um arquivo denominado ".profile", que é um arquivo de comandos (shell). O shell executará todos os comandos deste arquivo, assim que o usuário acessar um diretório inicial onde ele estiver.

Um diretório é a área de um disco onde são armazenados os nomes dos arquivos e algumas características como data da criação, tamanho em bytes, e outras. Pode-se ter vários diretórios em um disco. Para se deslocar de um para outro, deve-se saber o percurso (caminho de um diretório a outro).

Para encerrar as atividades usa-se, em super-usuário, "desliga", que é o arquivo designado para este fim.

É importante ressaltar, embora seja comum na maioria dos manuais, que os símbolos "[" e "..." usados nas sintaxes dos comandos a seguir, indicam argumentos opcionais e que argumentos podem ser repetidos, respectivamente.

3. COMANDOS MAIS USADOS NO SISTEMA OPERACIONAL EDIX

Obs: os comandos do EDIX devem ser digitados em letras minúsculas. Os comandos `imp`, `sus`, `pro`, `can`, `pre`, `lib`, `pri`, `rem`, `sta` e `mos` são comandos de spooler (controle de impressora, controle de pedidos de impressão, status de impressoras/pedidos). Alguns comandos só podem ser usados em modo super-usuário.

`mkdir` - Usado para criar diretórios. Quando o sistema cria um diretório, automaticamente cria as entradas para os nomes `.` e `..`.

(`.` pseudônimo para o diretório atual)

(`..` pseudônimo para o diretório origem do diretório atual)

Sintaxe: `mkdir dir`

`mkdir` - comando

`dir` - nome do diretório que será criado

EX1: `mkdir dir`

Cria um diretório (`dir`), que será um subdiretório do diretório atual.

`rmdir` - Usado para remover diretórios vazios. As entradas `."` e `".."`, não podem ser removidas. Um diretório que contém somente estas entradas é considerado vazio. Para remoção de um diretório, deve-se posicionar no diretório pai do diretório a ser removido.

Sintaxe: `rmdir dir`

`rmdir` - comando

`dir` - nome do diretório a ser removido

EX1: `rmdir dir`

Remove o diretório `dir`

`cd` - Permite mover-se para um novo diretório

Sintaxe: `cd [dir]`

`cd` - comando

`dir` - nome do diretório (caminho)

EX1: `cd /usr/bin`

Vai para o diretório `/usr/bin`

EX2: `cd ..` (`cd` seguido de dois pontos)

Vai para o diretório origem do diretório atual

Obs: se o diretório corrente é `/dir1/dir2`, então o novo diretório corrente será `/dir1`.

EX3: cd

Vai para o diretório inicial, ou seja, o diretório da abertura de sessão.

ls - Lista o conteúdo de um diretório. A saída é em ordem alfabética. Os arquivos que iniciam com ponto (.) são listados somente com a opção a. Com as opções, ls dá várias informações sobre os arquivos.

Sintaxe: ls [-ltsdriu] [nome]

ls - comando.

Opções mais usadas:

- l - lista em formato longo, dando o tipo do arquivo, o modo, o número de ligações, o nome da conta do dono, o grupo, o tamanho em bytes, data e hora da última modificação e o nome do arquivo ou diretório.
- t - classifica a lista de arquivos de acordo com as datas de modificações do arquivo (Primeiro os mais recentes).
- a - lista todas as entradas, inclusive as que começam por ponto (..).
- r - inverte a ordem de listagem pelo nome ou pelo tempo de acesso.
- s - mostra o tamanho em blocos, para cada arquivo. (lb = lk).

Formato da listagem

trwxrwxrwx	_l_	_p_	_g_	_t_	mmm hh:mm	/arquivo
-----	:	:	:	:	:	:
:	:	:	:	:	:	:
1 2	3	4	5	6	7	8

1 - Tipos de arquivos, onde:

- - comum
- d - diretório
- b - bloco (arquivo especial)
- c - caracter (arquivo especial)
- p - pipe com nome fifo

2 - Proteção de arquivos em três níveis:

rwXrwxrwx

: --- :
 : : -----> outros usuários
 : -----> grupo de usuários
 -----> proprietário

r - permissão para leitura

w - permissão para gravação
 x - permissão para execução
 - - não tem permissão. Permissão negada

- 3 - Ligação para arquivos
- 4 - Proprietário
- 5 - Grupo
- 6 - Tamanho de arquivo em bytes
- 7 - Data da modificação (mês hora:minuto)
- 8 - Nome do arquivo

EX1: ls /usr

Lista os arquivos do diretório /usr (usuários).

EX2: ls -a

Lista todas as entradas inclusive as que iniciam com ponto.

EX3: ls -l

Lista em formato longo os arquivos do diretório corrente.

EX4: ls -t /bin

Lista os arquivos do diretório /bin de acordo com o tempo de modificação. Lista primeiro os modificados por último.

EX5: ls -rt *.doc

Lista os arquivos do diretório atual, cujos nomes terminam em .doc, com os arquivos modificados menos recentemente, listados primeiro.

cat - Programa para concatenar e imprimir arquivos. É um dos mais versáteis programas de manipulação de arquivos.

Sintaxe: cat [-us] [arq1] [>] [arq2]

cat - comando

u - opção usada para suprimir o armazenamento temporário de blocos utilizados pelo cat (Um bloco = 512 bytes).

s - força o cat a não se pronunciar sobre arquivos não existentes.

arq1 - nome do arquivo de entrada (opcional). Se não especificado, os dados serão lidos da entrada padrão.

> - significa a concatenação de arquivos. Neste caso, o arquivo de saída deve ser especificado. Também é opcional.

arq2 - nome do arquivo de saída.

Obs: não se deve usar o nome do arquivo de saída igual ao de en-

trada, quando se usa o cat para concatenar arquivos.

EX1: cat arq

Imprime no terminal o conteúdo do arquivo arq.

EX2: cat arq1 arq2 arq3

Imprime no terminal o conteúdo dos arquivos arq1 arq2 e arq3.

EX3: cat arq1 arq2 arq3 > arq4

Combina vários arquivos (arq1 arq2 arq3) no arquivo arq4.

EX4: cat /dev/null > novo

Cria um arquivo vazio chamado novo, no diretório atual.

EX5: cat > arq1

Fica em aberto para digitar alguma coisa. Próprio para pequenos arquivos. Finaliza com control d.

EX6: cat arq1 arq2 >> arq3

Cria o arquivo arq3 contendo os dados de arq1 seguido dos de arq2, se arq3 não existe. Caso exista, inclui os outros.

EX7: cat arq1 > /dev/tty2

Lista o conteúdo do arquivo arq1 no terminal tty2.

imp - Usado para imprimir arquivos em uma impressora (off-line). Pode ser usado com pipes.

Sintaxe: imp [-arg] [nome]

imp - comando.

Obs: imp sem nome de arquivo lê a entrada padrão.

arg - são os argumentos que podem ser usados. São eles: g, q, c, n, f, i.

n - número de cópias.

EX1: imp arq1

Imprime na impressora uma cópia do arquivo arq1.

EX2: imp -3 arq1

Imprime na impressora três cópias do arquivo arq1.

sus - Usado para suspender a impressão.

Sintaxe: sus

sus - comando.

EX1: sus

Suspende a impressão.

- pro** - Usado para dar continuidade à impressão suspensa pelo sus.
 Sintaxe: pro
 pro - comando.
 EX1: pro
 Prossegue a impressão.
- can** - Este comando é usado para cancelar uma impressão corrente numa impressora. Se o nome da impressora for omitido, assume a principal. Este comando se aplica à impressoras em estado de impressão.
 Sintaxe: can [-fn] [impressora]
 can - comando.
 f - encerra a impressão e desativa o servidor de impressão (spooler).
 n - desativa imediatamente o servidor de impressão.
 EX1: can
 Cancela a impressão corrente e procura outro pedido.
- pre** - Usado para prender pedidos de impressão. É melhor prender pedidos que ainda não estão sendo impressos, senão a impressão continua, embora preso, a não ser que a suspendemos. Com o comando lib, liberamos os pe didos presos.
 Sintaxe: pre no.pedido ...
 pre - comando
 no.pedido - é o número do pedido que será preso.
 EX1: pre 421
 Prende o pedido número 421.
- lib** - Usado para liberar pedidos de impressão, presos pelo comando pre.
 Sintaxe: lib no.pedido ...
 lib - comando.
 no.pedido - número do pedido que será liberado para impressão.
 EX1: lib 421
 Libera o pedido número 421.
- pri** - Comando usado para mudar a prioridade de pedidos de impressão. Atribue novas prioridades que são: (p, p+1, p+2, ...).
 Sintaxe: pri [-p] no.pedido ...
 pri - comando
 p - ordem de prioridade. Se p é omitido, a prioridade é um.

EX1: pri no.pedido

A prioridade do pedido passa a ser 1.

EX2: pri -20 508 507 506

A prioridade dos pedidos passa a ser a seguinte:

O pedido 506 passa para a prioridade 22

O pedido 507 passa para a prioridade 21

O pedido 508 passa para a prioridade 20

mos - Serve para mostrar ao usuário os pedidos de impressão.

Sintaxe: mos [-tp]

mos - comando

t - lista os pedidos de todos os usuários

p - lista na ordem dos pedidos

Obs: se opções forem omitidas, lista somente os pedidos do usuário dono

rem - Usado para remover pedidos da fila de impressão. O número dos pedidos é visto pelo comando mos.

Sintaxe: rem np ...

rem - comando

np = número do pedido

EX1: rem 235

Remove o pedido de impressão número 235

sta - Fornece o status das impressoras

Sintaxe: sta [impressora ...]

sta - comando

impressora - nome da impressora (instalação). Se omitida, fornece o status de todas

EX1: sta

Fornece o status das impressoras

cp - Copia arquivos (Em winchester)

Sintaxe: cp arq1 arq2

Sintaxe: cp arq1 [arq2...] destino

cp - comando

arq1 - nome do arquivo a ser copiado

arq2 - novo nome para a cópia (Primeiro caso) ou arquivo a ser copiado (Segundo caso)

destino - para onde os arquivos serão copiados.

O arquivo fonte não é alterado pela cópia. A cópia pode ser feita no mesmo diretório ou em diretórios diferentes (Especificar o caminho).

EX1: cp arq1 arq2

Copia arq1 com o nome de arq2. Arq1 não é afetado. Se arq2 já existir, seu modo e sua propriedade não serão afetados.

EX2: cp arq1 arq2 arq3 dir1

Copia os arquivos arq1 arq2 e arq3 no diretório dir1, se dir1 for subdiretório imediato do diretório atual

EX3: cp dir1/* dir2

Copia todos os arquivos do diretório dir1, para o diretório dir2.

EX4: cp *.com dir2

Copia todos os arquivos terminados com ".com" do diretório atual para o diretório dir2, se dir2 for subdiretório imediato do diretório atual.

mv - Usado para mover ou renomear arquivos.

Sintaxe: mv arq1 arq2

Sintaxe: mv arq1 [arq2...] destino

mv - comando

arq1 - arquivo a ser nomeado (Primeiro caso) ou arquivo a ser movido (Segundo caso)

arq2 - novo nome do arquivo (Primeiro caso) ou arquivo a ser movido (Segundo caso)

destino - para onde os arquivos serão movidos.

Em sua forma mais simples, mv muda o nome de um arquivo:

EX1: mv arq1 arq2

arq1 é renomeado para arq2

De outra maneira

EX2: mv dir1/* ../dir2

Move todos os arquivos do diretório atual dir1, para o subdiretório dir2 do diretório origem.

ln - Este comando serve para criar outros nomes para um arquivo existente. Embora existam vários nomes, há somente uma cópia dos dados no arquivo.

Sintaxe: ln arq1 [arq2]

Se arq2 for omitido, o pseudônimo será criado no diretório atual com o mesmo nome do último componente do nome de via que referencia arq1.

EX1: ln arq1 arq2

Cria o pseudônimo arq2 para arq1

rm - Programa para remover arquivos de um diretório. Os arquivos devem ter permissão de gravação no diretório

Sintaxe: rm [-fri] arq1...

rm - comando

f - opção de força. Independente de permissão de gravação no arquivo

r - opção recursiva, utilizada para remover uma subárvore inteira do sistema de arquivos. Pode-se usar a opção -i com -r para fazer com que a opção recursiva seja mais controlada

i - Pergunta se realmente se deseja remover cada arquivo (y ou n)

EX1: rm arq1

Remove o arquivo arq1

EX2: rm arq1 arq2 arq3

Remove os arquivos arq1 arq2 arq3

EX3: rm /dir1/*

Remove todos os arquivos do subdiretório dir1. Este é subdiretório do diretório raíz.

EX4: rm -i dir1/*

Remove todos os arquivos do subdiretório dir1, interativamente, se este for subdiretório imediato do diretório atual.

EX5: rm -r /dir1/dir2/dir3

Remove a subárvore liderada pelo diretório /dir1/dir2/dir3

chmod - Programa para mudar o modo de acesso a um arquivo ou diretório. Somente o proprietário ou super-usuário pode mudar o modo de um arquivo. Há três tipos de acesso: leitura, gravação e execução. Há também três categorias de usuários: proprietário, grupo e outros.

Sintaxe: chmod [ugoa] [+ rwxstugo] nome do arquivo

u g o a - indica para quem será alterada a proteção do arquivo.

QUEM

OPERADORES

u - usuário proprietário

+ - acrescenta permissão

g - grupo - - remove permissão
 o - outros usuários = - atribui permissões
 a - todos os usuários Obs: se omitidas, todas serão ativadas

PERMISSÕES

r - leitura
 w - gravação
 x - execução
 s - ajusta a identificação do usuário
 t - o modo de salvamento de texto
 u - permissão presente do usuário
 g - permissão presente do grupo
 o - permissão presente do outro

EX1: `chmod a+r arq1` ou `chmod +r arq1`

Acrescenta permissão de leitura ao arquivo arq1 para todos os usuários

EX2: `chmod go-rw arq1`

Torna o arquivo arq1 impossível de ser lido ou gravado por qualquer um que não seja o proprietário

EX3: `chmod a=rw arq1 arq2`

Torna arq1 e arq2 possíveis de serem lidos e gravados, mas não executáveis para todos

`chgrp` - Serve para alterar o grupo de um diretório ou arquivo. Usado em transferência de arquivos

Sintaxe: `chgrp ng arq ...`

`chgrp` - comando

ng - novo grupo do arquivo

arq - arquivo cujo grupo será trocado

EX1: `chgrp smq arq1`

Muda o grupo do arq1 para o grupo smq

chown - Serve para alterar o proprietário (dono) de um diretório ou arquivo.
Usado em transferências de arquivos

Sintaxe: `chown nd arq1 ...`

`chown` - comando

`nd` - novo dono do arquivo

`arq1` - arquivo cujo dono será trocado

EX1: `chown luiz *.doc`

Muda o dono de todos os arquivos tipo doc para o proprietário (dono) luiz.

file - Mostra o tipo de arquivo, determinando se o arquivo possui informações binárias ou de textos. Os arquivos de textos podem ser digitados no terminal

Sintaxe: `file arq ...`

`file` - comando

`arq` - arquivo cujo tipo será mostrado

EX1: `file *`

Determina os tipos de arquivos do diretório atual

EX2: `file /usr/bin/*`

Determina os tipos de todos os arquivos do diretório /usr/bin

EX3: `file arq1`

Determina o tipo do arquivo arq1 do diretório atual

tail - Mostra o final de um arquivo texto

Sintaxe: `tail [+num[lbc]] arq...`

- - deslocamento relativo ao final do arquivo

+ - determina que n linhas, caracteres e/ou blocos serão saltados a partir do início do arquivo

num - número de linhas, blocos e/ou caracteres

l - linhas

b - blocos

c - caracteres

Obs: se omitidos, assume linhas

EX1: `tail arq1`

Imprime as últimas 10 linhas do arquivo arq1

EX2: `tail -200c arq1`

Imprime os últimos 200 caracteres do arquivo arq1

EX3: tail +500 arq1

Imprime o arquivo arq1 a partir da linha 501

- df - Usado para informar o número de blocos livres em um sistema de arquivos ativos. Se o sistema de arquivos não for mencionado, informa o espaço livre de todos os sistemas de arquivos montados

Sintaxe: df [-tf] [sist.arqs]

df - comando

t - informa o número total de blocos alocados

f - informa o número de blocos na lista livre

sist.arqs - sistema de arquivos tratado

EX1: df

Informa o número de blocos livres dos sistemas e i-nodos.

- du - Fornece um resumo da utilização do disco. Dá o espaço utilizado pelos arquivos e o total utilizado pelo diretório (Espaço em blocos)

Sintaxe: du [-as] [arq ...]

du - comando

Opções:

a - informa o número total de blocos para cada arquivo

s - não informa para os arquivos especificados. Informa o total de blocos

arq - qualquer arquivo ou arquivos

EX1: du -a arq1 arq2

Informa o número de blocos dos arquivos arq1 e arq2

- wc - Conta o número de linhas, palavras e caracteres existentes nos arquivos tipo texto e imprime-os no vídeo

Sintaxe: wc [-opções] [arq...]

wc - comando

Opções:

l - serve para direcionar a contagem por linhas

w - serve para direcionar a contagem por palavras

c - serve para direcionar a contagem por caracteres

Obs: se não especificadas, assume (-lwc)

arq - nome do arquivo submetido ao comando

EX1: wc arq1 ou wc -lwc arq1

Conta as linhas, palavras e caracteres do arquivo arq1

EX2: wc -w arq1

Conta o número de palavras do arquivo arq1

EX3: ls | wc -l

Conta o número de arquivos no diretório corrente

- od - Usado para listar o conteúdo de um arquivo em ASCII, decimal, hexadecimal ou octal, de acordo com as opções utilizadas. Se o nome do arquivo for omitido lê a entrada padrão

Sintaxe: od [opções] [arq]

Opções:

- c - lista os bytes em ASCII
- d - lista palavras em decimal
- x - lista palavras em hexadecimal
- b - lista bytes em octal
- o - lista palavras em octal
- arq - arquivo que será lido

EX1: od arq1 ou od -o arq1

Lista o arquivo arq1 no formato de palavra octal

EX2: od -x arq1

Lista o arquivo arq1 no formato de palavra hexadecimal

- find - Percorre o sistema de arquivos do início para o final, procurando os arquivos que atendam as condições especificadas

Sintaxe: find [percurso] [condição]

find - comando

percurso - nome do percurso do sistema de arquivos

condição - várias

Condições mais usadas:

exec cmd - especifica um comando a ser executado

user nome - especifica o proprietário de um arquivo

print - gera o nome da via atual

EX1: find . -print

Imprime o nome dos arquivos da subárvore atual

EX2: find /usr/paulo -user jose -print

Encontra todos os arquivos da subárvore paulo que pertençam a jose e os imprime no vídeo

EX3: find /usr/paulo -user jose -print | imp

Idem ao EX2, só que a saída será na impressora.

diff - Programa usado para comparar dois arquivos ou diretórios. Serve para mostrar quais linhas são diferentes em dois arquivos. As linhas do primeiro arquivo são precedidas do símbolo < e as do segundo arquivo, do símbolo >

Sintaxe: diff [-efbh] arq1 arq2

Opções:

- e - produz saída na forma de comandos do editor. Compara os dois arquivos e mostra as linhas do segundo arquivo que são diferentes das linhas do primeiro
- f - produz um script semelhante a -e na ordem inversa. O script produzido por -f não pode ser usado pelo editor padrão do UNIX
- b - ignora as diferenças entre as linhas causadas pelos espaços em branco (espaços e tabulações)

Pseudocomandos do editor gerados pelo diff:

n1 a n3, n4

n1, n2 c n3, n4

n1, n2 d n3

Os números de linhas n1 e n2 referem-se ao primeiro arquivo e n3 e n4 ao segundo.

n1 a n3, n4 - indica que o segundo arquivo contém linhas (n3 até n4), que não estão no primeiro arquivo após n1

n1, n2 c n3, n4 - indica que as linhas n1 até n2 no primeiro arquivo são diferentes das linhas n3 e n4 no segundo arquivo

n1, n2 d n3 - indica que o primeiro arquivo contém linhas (n1 a n3) que estão faltando no segundo arquivo após n3

EX1: se arq1 e arq2 são iguais exceto que arq2 contém a linha EMBRAPA

- CNPSA após a linha 10, então o comando:

```
diff arq1 arq2
```

irá produzir a saída:

```
10all
```

```
> EMBRAPA - CNPSA
```

comm - Compara dois arquivos ordenados, procurando as linhas em comum

Sintaxe: comm [-123] arq1 arq2

Opções: este comando permite usar 3 opções, a seguir:

- 1 - não deixa imprimir a primeira coluna (linhas de arq1)
- 2 - não deixa imprimir a segunda coluna (linhas de arq2)
- 3 - não deixa imprimir a terceira coluna (linhas comuns)

Obs: se as opções forem omitidas, imprime, então, as três colunas

EX1: comm arq1 arq2

Compara arq1 e arq2 imprimindo todas as colunas (3)

EX2: comm -12 arq1 arq2

Compara arq1 e arq2 imprimindo apenas as colunas comuns aos dois arquivos

split - Comando usado para dividir arquivos em trechos de n linhas em um conjunto de arquivos (Omissão = 1000)

Sintaxe: split [-n] [arq1 [arq2]]

split - comando

-n - número de linhas que serão contadas no trecho

arq1 - arquivo de entrada, e que será dividido em trechos. Se omitido, ou se for utilizado "-", é assumida a entrada padrão

arq2 - arquivo de saída. Será acrescido de sufixo(s)

O split acrescentará sufixos aos nomes dos arquivos. Criará tantos arquivos quantos forem necessários. Na ausência de arq2, x é assumido

EX1: seja o arquivo arq1 com o seguinte conteúdo:

caneta

lapis

borracha

caderno

bloco

livro

usando: split -3 arq1 arq2

O arquivo arq1 será dividido em trechos de três linhas e gravado em arq2aa e arq2ab. O arquivo arq2 não existe.

O conteúdo de arq2aa será:

caneta

lapis

borracha

O conteúdo do arq2ab será:

caderno

bloco

livro

EX2: split -5 arq2

Assume a entrada padrão. Deve-se digitar o arquivo após o comando acima e ele será dividido em arquivos com 5 linhas cada (arq2ab, arq2ac, ...), dependendo logicamente do número de linhas digitadas

tee - O programa tee duplica a entrada padrão (STDIN). Lê a entrada padrão e desvia para a saída padrão. Entre outras utilidades, é usado para se obter resultados intermediários numa canalização, desviar a saída de um programa para um arquivo, ...

Sintaxe: tee [-ia] [arq] ...

tee - comando

i - faz tee ignorar as interrupções

a - faz tee adicionar a saída aos arquivos

EX1: tee arq1

Usa arq1 para gravar a entrada padrão na saída padrão

EX2: cat arq1 | tee arq2

Lista o conteúdo de arq1 e grava o mesmo em arq2

ps - Utilizado para dar informações sobre os processos ativos. Este comando, sem opções, imprime o número do processo, nome do tty de controle, o tempo de execução acumulado e uma aproximação da linha de comando. Estas informações referem-se ao terminal corrente

Processos:

São programas em estado de execução. Apresentam-se em dois tipos de estado:

vanguarda - executado interagindo com o usuário

retaguarda - executado sem a ação do usuário. A vantagem de usar processos em retaguarda é a liberação imediata do terminal

Sintaxe: ps [opções]

Opções mais usadas:

l - gera uma lista em formato longo

e - imprime informações de todos os processos

f - gera uma lista completa

EX1: ps

Gera uma lista dos processos atuais

- nohup** - Usado para executar programas imunes a desligamentos. Estes não se perdem quando se sai do sistema. Se não se utiliza um redirecionamento de saída, esta será gravada em um arquivo chamado nohup.out.
 Sintaxe: nohup processo [arg]
 nohup - comando
 processo - processo que será executado
 arg - argumentos (retaguarda (&), redirecionamento, ...)
 EX1: nohup prog1 &
 Processa o prog1 em retaguarda
 EX2: nohup prog1 > prog2 &
 Executa o processo prog1 em retaguarda. Saída em prog2
- who** - Lista o nome, o terminal, mês, dia, hora e início da sessão de um ou mais usuários que estão acessados ao sistema
 Sintaxe: who [am I]
 EX1: who
 Lista os usuários atuais
- pwd** - Mostra o nome do diretório corrente. Imprime o nome completo da via deste diretório
 Sintaxe: pwd
 EX1: pwd
 Mostra na tela o nome do diretório corrente
- date** - Comando utilizado para exibir a data e hora
 Sintaxe: date [mmddhhmm[yy]]
 EX1: date
 Exibe a data e a hora no vídeo
- time** - Serve para controlar o tempo de um processo. Após o processo completar-se, aparecerá no vídeo o tempo total decorrido, tempo de execução do processo e o tempo dispendido pelo comando
 Sintaxe: time processos [arg]
 time - comando
 processos - nome do processo (ou processos) a ser executado
 arg - qualquer argumento para complementação do comando, por exemplo, redirecionar saída, produzir listagem, etc.

EX1: time prog1

Fornece o tempo do processo prog1

EX2: time who

Marca o tempo do processo who

EX3: time ls -l /bin

Fornece o tempo produzido pela listagem de formato longo do diretório /bin

EX4: time ls -l /bin &

Idem EX3, mas processa em retaguarda

kill - Usado geralmente para terminar um processo em retaguarda (apagar)

Sintaxe: kill [-sinal] np

kill - comando

sinal - sinal enviado pelo comando

Obs: o sinal 9 termina incondicionalmente um processo

EX1: kill -9 345

Apaga o processo de número 345

sleep - Suspende a execução de um processo por um determinado período de tempo em segundos (0 tempo deve ser inferior a 65.536 segundos)

Sintaxe: sleep tempo; comando

sleep - comando

tempo - tempo de suspensão em segundos

EX1: sleep 120; cat arq1 > /dev/tty2

Aguarda 120 segundos antes de executar o comando cat

tty - Serve para imprimir o nome de arquivo especial do terminal

Sintaxe: tty

EX1: tty

Imprime o nome do terminal em uso

stty - Comando de ajuste de opções no terminal. Controla estas opções para que o sistema possa manipular o terminal corretamente. Pode-se ajustar a taxa de baud (... 1200, 1800, 2400, ...)

Sintaxe: stty (opções)

As opções são várias, porém apenas algumas são citadas:

erase c - ajuste de caracter para apagar c (c é qualquer caracter do teclado)

echo - Ecoa (não ecoa) na tela cada carácter no momento em que ele é digitado

EX1: stty

Mostra as opções atuais

EX2: stty página (ou +página)

Mantém a paginação na tela, aguardando pressionar uma tecla quando esta enche, para passar à próxima

sort - Ordena ou intercala arquivos de acordo com opções. Ordena linhas de arquivos de entrada e então escreve-as reorganizadas na saída padrão

Sintaxe: sort [-opções] [+pos1 [-pos2]]...[-o saída] [nome]

sort - comando

Opções mais usadas:

-b - ignora brancos no início da linha

-d - ordem de dicionário (letra/número)

-f - trata todas as letras como minúsculas

-n - ordena números pelo seu valor

-r - classifica em ordem inversa

+pos1 - posição inicial da classificação (Número de campos a serem saltados do início). Omissão: assume 0 (Coluna 1).

-pos2 - posição final da chave de classificação. Omissão: assume até o fim da linha

-o saída - grava o arquivo ordenado em um arquivo indicado (-o saída)

Obs: se nenhuma opção for usada, sort assume:

Chave de classificação - toda a linha

Ordem de classificação - seqüência de caracteres da máquina

EX1: sort arq1

Classifica (ordena) o arquivo arq1 com saída na saída padrão (vídeo)

EX2: sort arq1 > arq2

Classifica arq1 e coloca a saída em arq2

EX3: sort < arq1 > arq2

Idem ao EX2

EX4: sort -o arq2 arq1

Classifica o arquivo arq1 e grava a saída em arq2

EX5: sort -f arq1

Classifica arq1 sem considerar a forma maiúscula

EX6: sort -d arq1

Classifica arq1 na ordem de dicionário

EX7: seja o arquivo arq1 contendo:

luiz 441594 440070

paulo 440695 440122

vicente 442905 440070

jose 440345 440123 e fazendo:

sort +0 -1 arq1 > arq2

Classifica arq1 de acordo com o primeiro campo e grava o resultado em arq2

EX8: seja o arquivo arq1 do EX7 e fazendo:

sort +2 arq1

Classifica o arquivo arq1 de acordo com o 3º campo e lista no vídeo

EX9: sort +.5 -.10 arq1

Classifica arq1 da coluna 5 à 10 (0 ponto indica que é por coluna)

passwd - Muda ou cria uma senha de login (acesso)

Sintaxe: passwd ident

passwd - comando

ident - identificação do usuário

EX1: passwd

Permite alterar a senha do usuário corrente

mount e - Usados para montar e desmontar um sistema de arquivos. Montar e desmon

umount tar quer dizer vincular um sistema de arquivos e um meio físico, e des vincular, respectivamente. Um sistema de arquivos desmontado não permite acesso a ele. Não podemos usar estes comandos para o sistema raiz

Sintaxe: /etc/mount [dir.esp] dir2

/etc/umount dir.esp

EX1: umount /dev/wch2

Desmonta o sistema de arquivos /dev/wch2

EX2: mount /dev/wch2 /wch2

Monta o sistema /dev/wch2

echo - Imprime os caracteres contidos no comando após a palavra echo, na saída padrão. É utilizado para vários propósitos, como a inserção de pequenas quantidades de dados conhecidos numa canalização ou arquivo

Sintaxe: echo [arg]

echo - comando

arg - cadeia de caracteres que serão impressos na saída padrão (vídeo)

EX1: echo EMBRAPA

Mostra na tela a palavra EMBRAPA

EX2: echo EMBRAPA CNPSA > arq

Coloca a mensagem EMBRAPA CNPSA no arquivo arq

EX3: echo \$PATH

Mostra o valor da variável shell \$PATH

cal - Usado para imprimir um calendário

Sintaxe: cal [mês] ano

call - comando

Obs: mês (1 a 12), Ano (1 a 9999)

EX1: cal 10 1800

Mostra o mês de outubro (10) de 1800

EX2: cal 5 1988

Mostra o mês de maio (5) de 1988

banner - Imprime um argumento de até 10 caracteres em letras grandes, na saída padrão

Sintaxe: banner arg

banner - comando

arg - caracteres que serão impressos (máximo 10)

EX1: banner CURSO

Imprime a palavra CURSO em letras grandes

write - Envia mensagem para outro usuário. A qualquer hora que se envie uma mensagem, esta aparecerá na tela, desde que o terminal indicado esteja no ar, e em msg y

Sintaxe: write nome

write - comando

nome - nome do usuário ou número do terminal

EX1: write jose

texto

control d (para encerrar o texto)

Imprime o texto no terminal do usuário jose

Obs: se não se desejar receber mensagem através deste comando, usa-se mesg n. Isto inibe a recepção de mensagem. Com mesg y volta-se a receber

mail - Serve para enviar correspondência ou ler a nossa própria correspondência. Pode-se enviar para vários usuários simultaneamente

Sintaxe: mail usuário

Sintaxe: mail [-rpq] [-f arq1]

mail - comando

usuário - nome do usuário que vai receber a mensagem

Opção:

r - mostra as mensagens na ordem em que recebe

p - imprime as mensagens, uma após a outra, sem pausas

q - qualquer interrupção faz com que termine o mail

f arq1 - o arquivo arq1 é a fonte da correspondência

EX1: mail paulo

texto

control d

Envia a mensagem texto para o usuário paulo. O sistema envia ao terminal de paulo o aviso: "você tem correspondência". Então paulo deve digitar mail para recebê-la

Obs: opções fornecidas após a interrogação (?):

p - mostra a primeira mensagem

d - remove a mensagem corrente e mostra a próxima

4. INTERPRETADOR SHELL

O shell é considerado o utilitário mais importante do EDIX, é o interpretador de comandos. Fica disponível quando ativamos o sistema (quando ligamos a máquina). Qualquer comando, palavra ou caracter digitado, é interpretado por ele. Podemos ver isto, fazendo assim no terminal, logo após o prompt (\$), como uso interativo. A saída padrão de um comando pode ser conectada à entrada padrão de outro. Executa comandos lidos de um terminal ou arquivo.

Podemos também usar o shell como interpretador de linguagem de programação própria estruturada, envolvendo o uso de comandos compostos, como os laços condicionais while, until, if e laços de for, etc, permitindo ainda o uso de variáveis para armazenar valores. Elas podem ser ajustadas através de um comando de atribuição (v = EDIX, m = "Setor de Informática").

O shell possui cinco variáveis automáticas a saber:

- a) a variável \$? contém o valor retornado pelo último comando executado;
- b) a variável \$\$ contém o número do processo do shell;
- c) a variável \$! contém o número de processo do último processo invocado pelo shell;
- d) a variável \$# contém o número de argumentos para o shell (parâmetros posicionais);
- e) a variável \$- contém os sinalizadores que foram passados para o shell (Exs: echo \$\$ echo \$?).

Temos os parâmetros posicionais, que são observados pela posição que ocupam na linha de comandos. Variam de \$0, \$1, \$2, ... No shell usamos comandos simples (comandos e seus argumentos), canalizações (comandos simples ou grupo de comandos simples conectados) e listas (seqüências de canalizações). Na lista, os elementos são separados por ponto e vírgula.

Temos três maneiras de executar um programa shell. Por redirecionamento de entrada (sh < arq1), o arquivo sendo um argumento (sh arq1), e criando um arquivo texto e acertando seu modo de execução.

Por ser um assunto longo e relativamente complexo, veremos apenas alguns comandos do shell, considerados mais usados, a seguir:

- set - Usado para gerar uma lista de variáveis (lista o ambiente) e controlar vários modos internos do shell
- Sintaxe: set [-opções [arg]]
- set - comando

Opções mais comuns:

-v - imprime as linhas do shell, à medida que são lidas. Modo prolixo

+v - desativa o modo prolixo

* - os nomes dos arquivos do diretório corrente são atribuídos aos parâmetros posicionais. Atribua a \$1 o primeiro arquivo, a \$2 o segundo, e assim sucessivamente até o último arquivo do diretório

arg - pode ser nome de arquivo

Obs: sem opções e argumentos produz uma lista de variáveis criadas durante o acesso ao sistema

EX1: set

Lista as variáveis criadas no acesso ao sistema

EX2: set *

Lista os nomes dos arquivos do diretório corrente

test - Este comando é usado para testar condições (V ou F)

Sintaxe: test [-opções] arg

test - comando

Opções:

d - testa se o argumento seguinte é um arquivo diretório

t - testa se o argumento seguinte existe e é um arquivo

r - testa se o argumento seguinte é legível (permissão leitura)

w - testa se o argumento seguinte tem permissão de gravação

x - testa se o argumento seguinte tem permissão de execução

arg - qualquer argumento (condição a ser testada)

EX1: test -d /usr/luiz && echo ok

Testa se /usr/luiz é diretório. Se for, imprime ok

EX2: test -r arq1

É verdadeiro se o arquivo arq1 existir e tiver permissão de leitura

read - Este comando lê uma linha da entrada padrão. É usado para programação em shell. Associa a primeira variável ao primeiro nome e assim por diante

Sintaxe: read [var...]

read - comando

var - nome da variável que será lida

EX1: read op

Vai ler um valor para a variável op

shift - Este comando faz com que um ou mais parâmetros posicionais se desloque de uma posição, para trás (\$2 para \$1)

Sintaxe: shift

EX1: Considerando que arq1 contém:

```
ls $1; shift
```

```
cat $1; shift
```

ls \$1, e tendo três arquivos textos (arqa, arqb, arqc). Fazendo:

```
sh arqa arqb arqc, resulta:
```

```
lista o nome do arquivo arqa
```

```
lista o conteúdo de arqb
```

```
lista o nome de arqc
```

if - O if condicional do shell se comporta de maneira semelhante às linguagens de programação. Serve para testar determinadas condições

Sintaxe: if comandol

```
then comando2
```

```
[else comando3]
```

```
fi
```

EX1: supondo que arq1 contenha:

```
if test -d job1
```

```
then echo "e diretorio"
```

```
else echo "nao e diretorio"
```

```
fi
```

Testa se job1 é arquivo diretório ou não. Se for diretório, imprime "e diretorio", caso contrário imprime "nao e diretorio"

while - O while condicional permite repetir uma série de comandos, enquanto obedece determinada condição

Sintaxe: while comandol

```
do comando2
```

```
done
```

EX1: set *

```
while test -r $1
```

```
do echo "$1"; echo "legivel"
```

```
shift
```

done

Lista o nome de cada arquivo do diretório que tem permissão de leitura, seguido da palavra legível

until - Este comando é uma variação do while. O processo se repete caso a condição for falsa

Sintaxe: until comando1
do comando2

done

```
EX1: set *
      until test -w $1
          do ls -l $1; shift
```

done

Lista em formato longo o nome dos arquivos do diretório que não tem permissão de leitura

for - O for deixa um grupo de comandos ser executado (Um de cada vez)

Sintaxe: for var [in pal1 pal2 ...]
do comandos

done

```
EX1: for x in *
      do
          rm -i $x
```

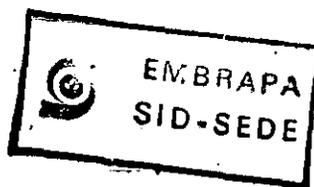
done

Este programa faz a leitura de todos os arquivos do diretório corrente e questiona se deve excluir cada arquivo

```
EX2: for arq in *
      do
          if test -d $arq
              then echo "$arq"; echo "e diretorio"
              else echo "$arq"; echo "nao e diretorio"
          fi
```

done

Lê todos os arquivos do diretório, listando o nome e, se é ou não diretório



case - case é um desvio múltiplo. Executa a lista associada ao primeiro padrão que case com var (abaixo):

```
Sintaxe: case var in
            [padrão1) comandos;;]
            [padrão2) comandos;;]
            [padrão3) comandos;;]
            ...
        esac
```

EX1: supondo que arq1 contém:

```
echo "\032"
while true
do
echo "Informe opcoes:"
read opcao
case $opcao in
    1) prog1;;
    2) echo "OPCAO 2";;
    3) echo "OPCAO 3";;
    9) break;;
    *) echo "opcao invalida"; sleep 5;;
esac
```

e prog1 seja um programa executável, fazendo:

```
sh arq1
```

Lista a tela e pede opções. Se opção=1, executa prog1, se opção=2 ou 3, lista respectivamente OPCA0 2 e OPCA0 3. Se opção=9 encerra o programa e volta ao sistema

break - Força a saída do laço corrente (for,while), se houver. Sai incondicionalmente do laço

Sintaxe: break [n]

Se n existir, sairá do n-ésimo nível do laço

continue- Desvia para a próxima interação do laço em uso

Sintaxe: continue [n]

Se n existir, então continua no n-ésimo nível do laço em uso

EX1: for arq in *

```
do
```

```

if test -d $arq
    then echo "$arq";break
else
    continue
fi
done

```

Lista os arquivos do diretório testando-os se são arquivos diretórios ou não. Quando encontra o primeiro (diretório), lista o seu nome e pára.

5. ED

O ED, editor de textos do EDIX, é um editor interativo, permite que se digite ou edite um texto e possibilita, além de outras facilidades, a inclusão, alteração ou deleção de palavras, linhas ou parágrafos. A importância na edição de textos está na facilidade de manuseio dos mesmos. É um método prático de entrada de dados.

O ED trabalha com modo de comando e modo de entrada de texto.

São apresentados apenas os principais comandos ou os mais usados. Para entrar no editor, deve-se teclar: ed [nomearq]. Os caracteres * e ? indicam que o ED está pronto para receber comando ou indicando erro de entrada de comando, respectivamente.

A melhor maneira de fixar é criar um arquivo e testar os comandos abaixo.

Comandos do editor de textos:

P - ativa (desativa) o pronto (*)

H - ativa (desativa) modo mensagem

h - exibe a última mensagem de erro encontrada

a - ativa a entrada de texto (No início ou no meio do texto). Ao digitar um ponto (.) encerra a entrada de texto e volta ao modo comando

w - comando usado para gravar o texto em um arquivo (w[arq])

p - exibe o conteúdo do buffer no terminal (texto):

n1[,n2]p

n1 - linha inicial

n2 - linha final

Se n2 for igual a \$, então imprimirá de n1 até o fim

r - lê o arquivo para o final do buffer (texto)

d - serve para deletar linhas

nl[,n2]d

q - sai do editor. O arquivo não é gravado automaticamente. Deve-se usar antes o comando w

Q - abandona o arquivo sem salvar. Não avisa

i - comando para inserir texto (insere antes do número da linha indicado). O ponto (.) encerra o modo inserção

m - serve para mover uma parte do texto (grupo de linhas). Marca-se o número da linha inicial e final do bloco a ser movido

t - serve para copiar linhas. Marca-se o número inicial e final de bloco a ser copiado

f - dá o nome do arquivo editado

\ - esta barra tira o significado especial dos caracteres

s - substitue seqüência de caracteres. Os caracteres / , . # + [] são usados para separar um ou mais caracteres da seqüência a ser substituída

g - executa um comando em todo o texto

6. INFOWORD (iw)

É um processador de textos fornecido pela INFOCOM Software, similar nacional ao WordStar, muito conhecido e relativamente simples. Roda nos sistemas operacionais UNIX (EDIX), MS-DOS, CP/M e MSX. Serve para criar arquivos e permite ser alterado em todos os sentidos, como correção de digitação errada, inserir textos, modificar margens, etc., após, gravar em disco, e então imprimir seus conteúdos tantas vezes quanto necessário. Trabalha com uma cópia do arquivo, para segurança. O texto anterior às alterações será gravado com o nome e tipo b1w, enquanto que o atual, só com o nome.

O INFOWORD possui vários menus para ajudar o usuário, e podem ser chamados a qualquer momento, um por um, quando se edita um arquivo.

São eles:

MENU DE ABERTURA - Surge assim que se chama o editor

MENU PRINCIPAL - Apresenta comandos para movimentar o cursor, edição e outros menus

MENU DE ACELERAÇÃO - (CTRL Q) - Mostra comandos para se deslocar rapidamente dentro de um arquivo em edição

MENU DE BLOCOS - (CTRL K) - Mostra comandos para manipulação de blocos

MENU DE IMPRESSÃO - (CTRL P) - Mostra comandos para impressão

MENU DE TELA - (CTRL O) - Mostra comandos para acertos de tela (Margens, espaçamento entre linhas, etc.)

MENU DE SOCORRO - (CTRL J) - Dá explicações gerais sobre o sistema

Para chamá-lo, tecla-se iw e return. Surge na tela o menu de abertura. Tem-se uma série de comandos, dentre eles D e N. São dois caminhos para criarmos um arquivo:

D - cria um arquivo formatado pelo editor (arquivo texto)

N - cria um arquivo não formatado (Arquivo não texto)

Ao escolher D ou N, surge na tela uma pergunta solicitando o nome do arquivo a editar, e logo após digita-se um nome. Surge então o menu principal, mostrando uma série de comandos, e logo abaixo estará o cursor, e então tudo estará pronto para digitar o arquivo. Aqui pode-se criar o arquivo, alterar, chamar os menus enfim, fazer qualquer tipo de manipulação no arquivo. Após isto, pode-se teclar CTRL K juntamente com S, D, X ou Q, caso se queira, respectivamente, gravar e permanecer no arquivo, gravar e sair do arquivo, mas ficar no menu de abertura, gravar e voltar ao sistema operacional, ou abandonar a edição.

Ao editar um arquivo arql já existente, para qualquer alteração, gravando-o novamente, o editor criará um arquivo arql.biw, que terá o conteúdo do arquivo anterior a alteração e o arquivo alterado continuará com o nome original. Isto é feito automaticamente pelo editor.

ACENTUAÇÃO - Os textos editados pelo INFOWORD, podem ser devidamente acentuados, como se faz em máquinas de datilografia, desde que o teclado do computador tenha as teclas para acentuação e C cedilhado. Por exemplo, a palavra "página" acentua-se assim: p + acento agudo + agina.

7. COMANDOS DO INFOWORD

A seguir serão apresentados os comandos dos menus do editor. Para maiores informações, há necessidade de se recorrer a um livro que trate deste assunto.

COMANDOS DO MENU DE ABERTURA

- Comandos para o sistema:

L - serve para mudar de diretório

F - mostra o diretório (liga/desliga)

R - serve para executar um programa

- X - encerra sessão do editor, retornando ao sistema operacional
- ! - usado para executar programas SHELL
- Comandos para edição de arquivos:
 - D - para criar arquivos formatados pelo editor (arquivos textos)
 - N - para criar arquivos não formatados (arquivos não textos)
- Comandos para nível de ajuda:
 - H - define nível
 - 3 - mostra todos os menus
 - 2 - suprime somente o menu principal
 - 1 - suprime os outros menus
- Comandos de arquivos:
 - P - serve para imprimir arquivos
 - E - serve para renomear arquivos
 - O - serve para copiar arquivos
 - Y - serve para remover arquivos

COMANDOS DO MENU PRINCIPAL

- Movimentação do cursor:
 - CTRL S - caracter à esquerda
 - CTRL D - caracter à direita
 - CTRL A - palavra à esquerda
 - CTRL F - palavra à direita
 - CTRL E - sobe uma linha
 - CTRL X - desce uma linha
- Movimentação do texto:
 - CTRL Z - sobe linha
 - CTRL W - desce linha
 - CTRL C - sobe texto
 - CTRL R - desce texto
- Remoção:
 - CTRL G - remove caracter sob o cursor, retrocedendo o restante da linha
uma coluna
 - CTRL T - remove palavra à direita do cursor, retrocedendo o restante da li
nha, com o primeiro caracter para a posição do mesmo

CTRL Y - apaga a linha do cursor
DEL - remove caracter à esquerda do cursor

- Outros menus:

CTRL J - mostra menu de ajuda
CTRL K - mostra menu de bloco
CTRL O - mostra menu de tela
CTRL Q - mostra menu de aceleração
CTRL P - mostra menu de impressão

- Miscelânea:

CTRL I - tabulação (igual ao tab)
CTRL B - reformata parágrafos
CTRL V - inserção (liga/desliga)
CTRL L - repete busca
<CR> - fim de parágrafo
CTRL N - insere uma linha
CTRL U - interrupção

COMANDOS DO MENU DE ACELERAÇÃO (CTRL Q)

- Movimentação do cursor:

S - início de linha
D - fim de linha
E - vai para o topo da tela
X - vai para a base da tela
R - vai para o início do arquivo
C - vai para o fim do arquivo
B - vai para o início do bloco
K - vai para o fim do bloco
Z - desloca o texto uma linha para baixo, continuamente
W - desloca o texto uma linha para cima, continuamente
L - posiciona numa linha solicitada
0-9 - Encontra marcadores de posição

- Miscelânea:

F - encontra texto no arquivo e para
A - localiza o texto no arquivo e o substitui por outro citado
Y - remove linha à direita do cursor

DEL - remove linha à esquerda do cursor

Q - repetição de comando

COMANDOS DO MENU DE BLOCOS (CTRL K)

- Operações com blocos:

B - marca início de bloco

K - marca fim de bloco

H - esconde/mostra a marcação de bloco

C - copia o bloco

Y - remove o bloco

V - move o bloco

W - grava o bloco

N - modo coluna (liga/desliga)

0-9 - define marcadores de posições

- Operações com arquivos:

R - lê (insere) arquivos na posição do cursor

O - copia um arquivo enquanto estiver em edição

E - renomeia arquivos em edição

J - remove arquivos em edição

S - grava e retorna ao arquivo na posição do cursor

D - grava e não retoma edição

X - grava e sai do editor. Volta ao sistema operacional

Q - abandona a edição

! - executa um programa SHELL

Obs: para retornar ao menu principal, teclar a barra de espaço

COMANDOS DO MENU DE IMPRESSÃO (CTRL P)

- Par de comandos:

B - imprime em negrito forte

D - imprime em negrito

S - imprime sublinhado

X - imprime riscado

- Comando único:

H - imprime caracter sobre caracter anterior (sobrepõe caracter)

<CR> - imprime próxima linha sobre a atual (sobrepõe linha)

COMANDOS DO MENU DE TELA (CTRL O)

- Margens e tabulações:

- L - define a margem esquerda
- R - define a margem direita
- X - liberação de margens (liga/desliga)
- I - define tabulação
- N - apaga tabulação
- G - deslocamento temporário da margem esquerda
- F - define régua com linha atual

- Funções de linha:

- C - centraliza o texto de acordo com a linha régua
- S - define espaçamento entre linhas

- Comandos alternantes:

- W - transporte de palavras (liga/desliga)
- T - mostra linha régua (liga/desliga)
- J - alinhamento da margem direita (liga/desliga)
- D - mostra comandos de impressão (liga/desliga)
- P - mostra separador de páginas (liga/desliga)

Obs: para retornar ao menu principal, teclar a barra de espaçamento

COMANDOS DO MENU DE AJUDA (CTRL J)

- I - índice de comandos
- H - nível de ajuda desejado
- B - reformatação de parágrafo
- D - comandos-ponto e controle de impressão
- F - coluna de sinalizadores
- M - ajuda sobre margens, espaçamento, alinhamento e tabulação
- R - ajuda sobre linha régua
- S - ajuda sobre linha de estado
- V - ajuda sobre movimentação de texto (ou de bloco)
- P - ajuda sobre marcadores de posição
- L - localização e substituição automática

8. EXERCÍCIOS

- 1) Criar um subdiretório denominado nome1, a partir de /usr/cursol.
- 2) Criar um subdiretório denominado nome2, a partir de /usr/cursol.
- 3) Criar um subdiretório denominado nome3, a partir de /usr/cursol.
- 4) Ver se os diretórios foram criados, seus nomes, e o que eles contêm.
- 5) Remover o subdiretório /usr/cursol/nome3.
- 6) Deslocar-se para o subdiretório de origem do diretório atual.
- 7) Deslocar-se para o diretório de origem (abertura do sistema).
- 8) Posicionar-se no subdiretório /usr/cursol/nome1.
- 9) Criar um arquivo pelo cat, denominado arq1 (Apêndice B).
- 10) Criar um arquivo pelo cat, denominado arq2 (Apêndice B).
- 11) Criar um arquivo denominado arq3, contendo o diretório atual, usando ls.
- 12) Listar o diretório atual em formato longo.
- 13) Listar o conteúdo do arquivo arq2.
- 14) Juntar os arquivos arq1 e arq2 em um arquivo arq4.
- 15) Imprimir o arquivo arq1. Criar os arquivos arq6 e arq7 (Apêndice B).
- 16) Criar um arquivo denominado lsdír com o comando ls, contendo o diretório corrente e imprimir.
- 17) Colocar na fila de impressão os arquivos arq1, arq2, arq3, arq4, arq6 e arq7. Suspender e deixar prosseguir a impressão de alguns deles. Mostrar a lista de impressão, remover alguns pedidos, e dar o status da impressora.
- 18) Criar um subdiretório denominado sdir, a partir do diretório raiz e posicionar-se novamente em /usr/cursol/nome1.
- 19) Ver as permissões dos arquivos do diretório atual.
- 20) Retirar todas as permissões do arquivo arq1, para todos os usuários.
- 21) Acrescentar permissões de leitura, gravação e execução no arquivo arq1, para o usuário proprietário.
- 22) Acrescentar permissões de leitura e execução para o grupo, no arquivo arq1.
- 23) Criar dois arquivos quaisquer e alterar o seu grupo. A seguir alterar o seu proprietário. Apagar os arquivos.
- 24) Determinar que tipo (binário ou texto) é o arquivo arq1.
- 25) Listar os últimos 15 caracteres do arquivo arq2.
- 26) Mostrar o número de blocos livres dos sistemas de arquivos.
- 27) Ver qual o espaço em blocos utilizado pelos arquivos.

- 28) Copiar o arquivo arq1 no subdiretório /usr/cursol/nome2.
- 29) Copiar os arquivos arq2 e arq3 no subdiretório /usr/cursol/nome2.
- 30) Fazer uma cópia de arq1 com o nome de arq5 no mesmo subdiretório.
- 31) Mover o arquivo arq3 para o subdiretório /usr/cursol/nome2.
- 32) Remover (apagar) o arquivo arq5 do subdiretório atual, perguntando caso realmente se deseja apagar.
- 33) Copiar do subdiretório /usr/cursol/nome1 para o subdiretório sdir o arquivo arq1 com o nome de arqlbak.
- 34) Dizer quantas linhas e quantas palavras existem no arquivo arq2.
- 35) Listar o conteúdo do arquivo arq1 em ASCII.
- 36) Encontrar os arquivos que estão no subdiretório /usr/cursol/nome2 e que pertençam a /usr/cursol/nome1 e listar na impressora.
- 37) Comparar os arquivos arq6 e arq7 e interpretar as diferenças usando o comando diff.
- 38) Comparar os arquivos arq6 e arq7, usando o comando comm.
- 39) Dividir o arquivo arq7 em arquivos de duas linhas e nomes com prefixo pre.
- 40) O que acontece com: cat arqa | tee arqb.
- 41) Criar o arquivo arq8 (Apêndice B) e executá-lo em retaguarda.
- 42) Ver quais usuários estão acessados ao sistema.
- 43) Dizer qual o diretório corrente.
- 44) Ver a data de hoje e a hora atual.
- 45) Dar a lista dos processos atuais.
- 46) Dar o tempo do processo who.
- 47) Cronometrar o tempo de execução do processo ls -l.
- 48) Listar o conteúdo do arquivo arq6, mas antes aguardar 15 segundos.
- 49) Ver qual o nome do terminal em uso.
- 50) Quando se lista um arquivo grande, que comando usar para que haja parada de tela?
- 51) Idem ao anterior, mas deixar a tela correr até o fim.
- 52) Ordenar alfabeticamente o arquivo arq7.
- 53) Ordenar alfabeticamente o arquivo arq7, gravar a saída em arq9.
- 54) Mostrar no vídeo a palavra CNPSA, usando o comando echo.
- 55) Faça um calendário do mês de outubro de 1985.
- 56) Imprimir a palavra CNPSA na impressora usando letras grandes (poster).
- 57) Enviar mensagem para o usuário luiz, usando o comando write.

- 58) Digitar o arquivo arq2 (Apêndice B) pelo ED e usar os comandos do mesmo. Acrescentar linhas, substituir linhas, transpor linhas, alterar linhas, listar linhas, gravar o arquivo, sair do editor, chamar novamente o arquivo e alterar, enfim, usar vários comandos para fixação (livre).
- 59) Digitar o arquivo iwtest1 (Apêndice B) pelo INFOWORD em modo de arquivo formatado pelo editor (D).
- 60) Digitar o arquivo iwtest2 (Apêndice B) em modo de arquivo não formatado pelo editor (N).

APÊNDICE A

SÍMBOLOS E CARACTERES ESPECIAIS:

- * - substitue qualquer seqüência de caracteres
- ? - substitue qualquer carácter na posição de ?
- | - pipe - conecta a saída de um processo com a entrada de outro (conexão)
- < - redirecionamento de entrada
- > - redirecionamento de saída (criação de arquivos)
- >> - redirecionamento de saída usado para inclusão
- << - inclusão de dados
- & - executa processos em retaguarda (Background)
- && - conectivo lógico "e" (and)
- || - conectivo lógico "ou" (or)
- ;- usado para processar uma seqüência de comandos. É um separador de comandos
- () - usados para agrupar comandos
- { } - usados para cercar variáveis (Ex: echo \${var}xy)
- \$1,\$2... são parâmetros posicionais. Usados para variáveis que contêm argumentos
- [...] - substitue no nome de arquivos os caracteres entre colchetes
- [m-n] - semelhante ao caso anterior, sendo que os caracteres são dados pelo intervalo de m até n

APÉNDICE B

ARQUIVOS

9 - arq1

1 1 1 1 1 1 1
2 2 2 2 2 2 2
3 3 3 3 3 3 3
4 4 4 4 4 4 4
5 5 5 5 5 5 5
6 6 6 6 6 6 6
7 7 7 7 7 7 7
8 8 8 8 8 8 8
9 9 9 9 9 9 9
10 10 10 10 10 10 10

10 - arq2

1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9
10 10 10 10 10 10 10 10 10 10 10
11 11 11 11 11 11 11 11 11 11 11
12 12 12 12 12 12 12 12 12 12 12
13 13 13 13 13 13 13 13 13 13 13
14 14 14 14 14 14 14 14 14 14 14
15 15 15 15 15 15 15 15 15 15 15
16 16 16 16 16 16 16 16 16 16 16
17 17 17 17 17 17 17 17 17 17 17
18 18 18 18 18 18 18 18 18 18 18
19 19 19 19 19 19 19 19 19 19 19
20 20 20 20 20 20 20 20 20 20 20

30 - arq6

arquivo arq6
EDIX
INFOWORD
WordStar
CP/M

- arq7

arquivo arq7
EDIX
INFOWORD
WordStar
CP-M
MSX

41 - arq8

```
cat > arq8
sleep 50; echo "funcionou"
Ctrl D
```

45 - arquivo iwtest1

Este arquivo deve ter o conteúdo da página 6 desta apostila(Introdução).

Digitar este arquivo pelo INFOWORD, em modo de arquivo formatado pelo editor (D). Usar os comandos para negritar, sublinhar, centralizar, acentuar, espaçamento 2, regular margens (esq. e dir.), duplicar o texto, alterar o texto (corrigir, inserir palavras), tabular, Imprimir para verificar os efeitos dos comandos.

46 - arquivo iwtest2

Digitar o arquivo a seguir em modo de arquivo não formatado pelo editor (N). (Tipo SAS ou SOC).

COMMENT - - - - - INFORMAÇÕES DE IDENTIFICAÇÃO

UNIDADE : 0021000 - CNPSA

PROGRAMA: 011 - PNP SUINOS

PROJETO : ESTUDO DE SISTEMAS DE ARMAZENAMENTO DE MILHO PARA A PRODUÇÃO DE RAÇÕES EM PEQUENAS PROPRIEDADES PRODUTORAS DE SUÍNOS

ARQUIVO : P0050873 - PÁGINA 097 - DISQUETE UAM-42-001

TRABALHO: IDEM PROJETO

ANO : 1987

PESQUISADOR RESPONSÁVEL: PAULO ARMANDO VICTÓRIA DE OLIVEIRA

ANALISTA RESPONSÁVEL :

- - - - - VARIÁVEIS DE CLASSIFICAÇÃO- - - - -

T = TRATAMENTOS (COL 1)

- 1 - MILHO ARMAZENADO EM SILO A GRANEL (SG)
- 2 - MILHO ARMAZENADO EM SILO COM ESPIGA SEM PALHA (SE)
- 3 - MILHO ARMAZENADO EM PAIOL COM ESPIGA COM PALHA (PA)
- 4 - MILHO DOBRADO E ARMAZENADO NA LAVOURA (LV)

P = PONTOS DENTRO DOS TRATAMENTOS (COL 3)

- 1 - AMOSTRAS COLHIDAS A 20 cm DO FUNDO DO TRATAMENTO
- 2 - AMOSTRAS COLHIDAS A 60 cm DO FUNDO DO TRATAMENTO

M = TEMPO DE ARMAZENAMENTO (BIMESTRAL) (COL 7)

1 - INICIAL

2 - 2 MESES

3 - 4 MESES

4 - 6 MESES

5 - 8 MESES

6 - 10 MESES

----- VARIÁVEIS DE RESPOSTA -----

MS - MATÉRIA SECA (% Gs) (COL 7)

EB - ENERGIA BRUTA (kcal/kg) (COL 12)

PB - PROTEÍNA BRUTA /10 (%) (COL 17)

ET - EXTRATO ETÉREO /10 (%) (COL 22)

FB - FIBRA BRUTA /10 (%) (COL 26)

GI - GRAU DE INFESTAÇÃO DE INSETOS /10 (%) (COL 30)

PP - PERDA DE PESO /10 (%) (COL 34)

1 1 1 878 3952 974 414 181 15 13

1 1 1 877 3951 951 399 177 17 11

1 2 1 879 4017 943 578 233 14 9

1 2 1 880 4002 965 525 229 12

2 1 1 857 3859 883 337 193 15 13

2 1 1 857 3873 890 363 201 17 11

2 2 1 862 3933 844 401 170 14 9

2 2 1 862 4043 867 402 166 12

3 1 1 797 3648 1012 488 292 15 13

3 1 1 798 3660 999 459 286 17 11

3 2 1 811 3701 1014 465 256 14 9

3 2 1 809 3796 992 458 260 12

4 1 1 788 3551 857 467 192 15 13

4 1 1 788 3569 819 454 176 17 11

4 2 1 779 3540 890 480 204 14 9

4 2 1 778 3532 841 474 207 12

Obsl: este arquivo contém somente uma parte dos dados numéricos, visto ser o objetivo maior, o comentário. Foi gentilmente cedido pelo pesquisador Paulo Armando V. de Oliveira (CNPSA).

APÊNDICE C

TABELA ASCII

DEC	HEX	OCT	ASCII	TECLA
0	00	00	NUL	CTRL ©
1	01	01	SOH	CTRL A
2	02	02	STX	CTRL B
3	03	03	ETX	CTRL C
4	04	04	EOT	CTRL D
5	05	05	ENQ	CTRL E
6	06	06	ACK	CTRL F
7	07	07	BEL	CTRL G
8	08	10	BS	CTRL H
9	09	11	HT	CTRL I
10	0A	12	LF	CTRL J
11	0B	13	VT	CTRL K
12	0C	14	FF	CTRL L
13	0D	15	CR	CTRL M
14	0E	16	SO	CTRL N
15	0F	17	SI	CTRL O
16	10	20	DLE	CTRL P
17	11	21	DC1	CTRL Q
18	12	22	DC2	CTRL R
19	13	23	DC3	CTRL S
20	14	24	DC4	CTRL T
21	15	25	NAK	CTRL U
22	16	26	SYN	CTRL V
23	17	27	ETB	CTRL W

DEC	HEX	OCT	ASCII	TECLA
24	18	30	CAN	CTRL X
25	19	31	EM	CTRL Y
26	1A	32	SUB	CTRL Z
27	1B	33	ESC	CTRL [
28	1C	34	FS	CTRL \
29	1D	35	GF	CTRL]
30	1E	36	RS	CTRL ^
31	1F	37	US	CTRL _
32	20	40	SP	ESPAÇO
33	21	41	!	!
34	22	42	"	"
35	23	43	#	#
36	24	44	\$	\$
37	25	45	%	%
38	26	46	&	&
39	27	47	'	'
40	28	50	((
41	29	51))
42	2A	52	*	*
43	2B	53	+	+
44	2C	54	,	,
45	2D	55	-	-
46	2E	56	.	.
47	2F	57	/	/
48	30	60	0	0
49	31	61	1	1

DEC	HEX	OCT	ASCII	TECLA
50	32	62	2	2
51	33	63	3	3
52	34	64	4	4
53	35	65	5	5
54	36	66	6	6
55	37	67	7	7
56	38	70	8	8
57	39	71	9	9
58	3A	72	:	:
59	3B	73	;	;
60	3C	74	<	<
61	3D	75	=	=
62	3E	76	>	>
63	3F	77	?	?
64	40	100	⊙	⊙
65	41	101	A	A
66	42	102	B	B
67	43	103	C	C
68	44	104	D	D
69	45	105	E	E
70	46	106	F	F
71	47	107	G	G
72	48	110	H	H
73	49	111	I	I
74	4A	112	J	J
75	4B	113	K	K

DEC	HEX	OCT	ASCII	TECLA
76	4C	114	L	L
77	4D	115	M	M
78	4E	116	N	N
79	4F	117	O	O
80	50	120	P	P
81	51	121	Q	Q
82	52	122	R	R
83	53	123	S	S
84	54	124	T	T
85	55	125	U	U
86	56	126	V	V
87	57	127	W	W
88	58	128	X	X
89	59	131	Y	Y
90	5A	132	Z	Z
91	5B	133	[[
92	5C	134	/	/
93	5D	135]]
94	5E	136	-	-
95	5F	137	-	-
96	60	140	\	\
97	61	141	a	a
98	62	142	b	b
99	63	143	c	c
100	64	144	d	d
101	65	145	e	e

DEC	HEX	OCT	ASCII	TECLA
102	66	146	f	f
103	67	147	g	g
104	68	150	h	h
105	69	151	i	i
106	6A	152	j	j
107	6B	153	k	k
108	6C	154	l	l
109	6D	155	m	m
110	6E	156	n	n
111	6F	157	o	o
112	70	160	p	p
113	71	161	q	q
114	72	162	r	r
115	73	163	s	s
116	74	164	t	t
117	75	165	u	u
118	76	166	v	v
119	77	167	w	w
120	78	170	x	x
121	79	171	y	y
122	7A	172	z	z
123	7B	173	{	}
124	7C	174		
125	7D	175	{	}
126	7E	176	~	~
127	7F	177	DEL	ELIM

9. BIBLIOGRAFIA CONSULTADA

- CHRISTIAN, K. Sistema operacional UNIX. Rio de Janeiro, Campus, 1985. 362p.
- EDISA ELETRÔNICA DIGITAL S/A., São Paulo, SP. Cartão de Referência EDIX. São Paulo, s.d. 1v.
- EDISA ELETRÔNICA DIGITAL S/A., São Paulo, SP. EDIX = Sistema Operacional. São Paulo, 1987. 1v.
- EDISA ELETRÔNICA DIGITAL S/A., São Paulo, SP. Manual de referência EDIX. São Paulo, 1986. 1v.
- LOMUTO, A.N. & LOMUTO, N. Introdução ao UNIX. Rio de Janeiro. Livros técnicos e científicos, 1987. 217p. (Ciência de Computação).
- SAMPAIO, M.C.; MOURA, A. & SAUVÉ, J.P. Processamento de textos: MS/DOS & UNIX. 2ª ed. São Paulo, McGraw-Hill, 1987. 217p.