

EDITOR DE TEXTOS DO TSO

2-) COMANDOS BÁSICOS

2.1) Comandos de linha do editor

2.1.1) Comandos Básicos

I (Insert), **D** (Delete), **R** (Repeat)

I => Insere linha. Para inserir uma ou mais linhas para digitação de novos dados

D => Deleta linha. Para deletar uma linha, várias linhas, ou um bloco de linhas

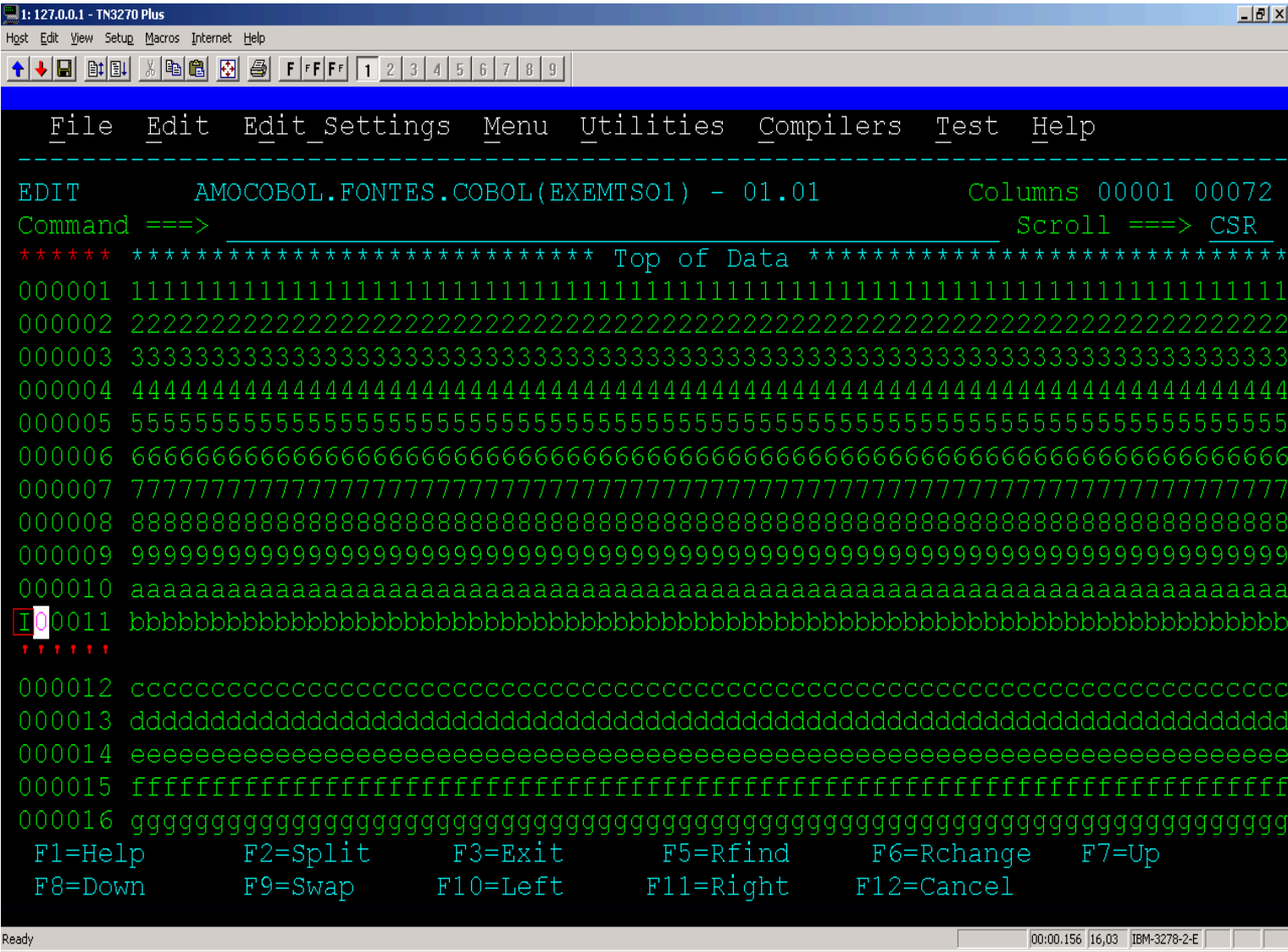
R => Repetição de linha. Para repetir uma linha, uma ou mais vezes, ou para repetir um bloco de linhas, uma ou mais vezes.

Exemplo 1:

000010

I00011 Uma linha será inserida após a linha 11

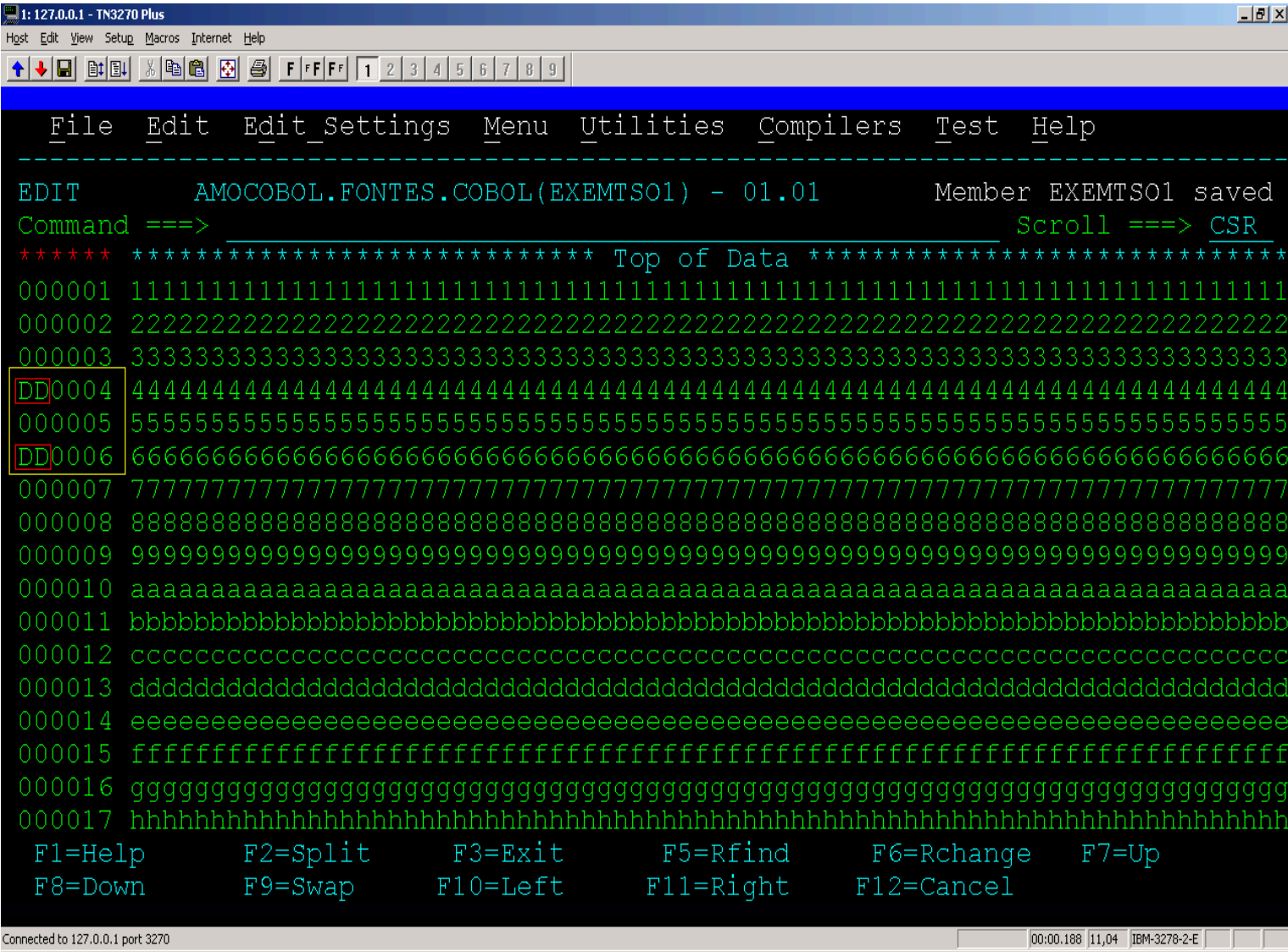
000002



000010

I30011 3 linhas serão inseridas depois da linha 11

000012



000001

D50002 5 linhas a partir da linha 2 serão deletadas

000003

Exemplo 3 :

R – Indica que uma única linha será repetida

R5 – Indica que a linha será repetida 5 vezes

RR – Identifica a primeira e a última linhas de um bloco de linhas a serem repetidas

RR2 – Identifica a primeira e última linhas de um bloco a ser repetido 2 vezes

000001

R30002 A linha 2 será repetida 3 vezes

000003

A cópia
ficará pendente até que seja dado o comando **A** (after) ou **B** (Before):

O comando de linha **A** (After) identifica o destino dos dados que serão movimentados ou copiados, isto é, serão movimentados ou copiados depois da linha onde foi executado o comando.

[illegible]

Resultado:

Copiar em Bloco

CC0001 O bloco composto pelas

000002 linhas 1, 2 e 3

CC0003 será copiado

- Os comandos primários COPY ou MOVE, se um membro de um arquivo particionado ou arquivo seqüencial será movimentado ou copiado para o texto sendo editado.

O comando de linha B (Before) identifica o destino para onde os dados movimentados ou copiados serão colocados, isto é, indica que os dados movimentados ou copiados serão colocados ANTES da linha contendo o comando.

[illegible]

O fonte dos dados para a operação de movimentação ou cópia podem ser especificados usando os mesmos recursos discriminados rio comando A (After).

O comando O (Overlay) indica a linha que será mergeada com as linhas copiadas ou movimentadas.

- O** – Identifica a linha que será mergeada
- O3** – Identifica a primeira de 3 linhas que serão mergeadas

Resultado

000001 /* */

000002

000003 A1 = B1; /* */

000004 A2 = B2; /* */

000005 A3 = B3; /* */

000006 H1 = A1 * B2;

2.1.3) Deslocamento

)
(Colunas a direita), ((Colunas a esquerda), > (Dados a direita),
< (Dados a esquerda)

)=> Desloca para a direita uma ou mais colunas

(=> Desloca para a esquerda uma ou mais colunas

>=> Desloca os dados a direita um ou mais caracteres

<=> Desloca os dados a esquerda um ou mais caracteres

Exemplos :

000002

)00003 Esta linha será deslocada para a direita 2 colunas

000004

)) Estas quatro linhas serão deslocadas para 000021 a direita 99 colunas, o que fará com que 000022 os dados destas 4 linhas sejam perdidos

))99

[illegible]

000006

(00007 Esta linha será deslocada para a esquerda 2 colunas 000500

< 0400 Dados nesta linha serão deslocados a esquerda 2 colunas

000500

ou

<< 700 Dados nestas 3 linhas serão 000800 deslocados a esquerda

<<4 00 pôr 4 colunas

001000

2.1.4) Exclusão/Exibiçãoooo

X (Exclude),
F (First), L (Last), S (Show)

X => Exclui uma linha, várias, ou um bloco de linhas

F => Mostra a primeira (ou primeiras) linha(s) de um bloco de linhas
excluídas

L => Mostra a última (ou últimas) linha(s) de um bloco de linhas excluídas

S => Mostra as linhas mais significativas de um bloco de linhas excluídas

Exemplos :

000300

X 0400 A linha 400 será excluída 000500

ou

XX 700 Este bloco 000800 de 3 linhas

XX 900 será excluído 001000

ou

000300

F - - - - - 10 LINE(S) NOT DISPLAYED

001400 (será exibida a primeira das 10 linhas acima)

F10 - - - - - 33 LINE(S) NOT DISPLAYED

005800 (serão exibidas as 10 primeiras das 33 linhas acima)

ou

000300

L ————— 10 LINE(S) NOT DISPLAYED

001400 (será exibida a última das 10 linhas acima)

L10 ————— 33 LINE(S) NOT DISPLAYD

005800 (serão exibidas as 10 últimas das 33 linhas acima)

ou

000300

S —————- 10 LINE(S) NOT DISPLAYD

001400 (será exibido uma das 10 linhas acima)

S10 —————- 33 LINE(S) NOT DISPLAYED

005800 (serão exibidas 10 das 33 linhas acima)

Obs.: Quando um grupo de linhas é excluído, o comando S pode ser usado

para exibir alguma, mas não todas as linhas . Para se determinar que

linhas serão exibidas, é usado indentação de dados. As linhas

indentadas mais a esquerda serão mostradas. Se várias linhas

forem indentadas de forma igual, isto é, com os dados começando

na mesma coluna), então, as primeiras linhas serão exibidas.

2.1.5) **Manipulação de texto**

TS (Text
split), TF (Text flow), TE (Text enter), LC (Lowercase),

UC (uppercase)

TS => Quebra de texto. Para quebrar uma linha de texto em duas a partir da
posição

do cursor; para permitir inserção de dados.

TF => Defragmentação de texto. Para juntar sentenças em um parágrafo.

TE => Entrada de texto. Para formatar a tela no sentido de facilitar a digitação de

parágrafos de texto.

LC => Caixa baixa. Para alterar o texto de letras maiúsculas para letras minúsculas.

UC => Caixa alta. Para alterar o texto de letras minúsculas para letras maiúsculas.

Exemplo do comando TS :

000300

TS 400 Esta linha será quebrada em duas para permitir

000500 inserção no texto

Resultado :

000300

000400 Esta linha será quebrada em duas

000410 para permitir

000500 inserção no texto

Obs.: Um número seguindo o comando TS indica o número de linhas a serem inseridas entre as duas partes do texto.

Pôr exemplo, o comando TS3 fará cm que sejam inseridas 3 linhas

entre as duas partes do texto.

Exemplo do comando TF :

TF 400 Este texto

000500 composto pôr um conjunto de linhas, pode ser 000600 reformatado de forma a tornar

000700 o parágrafo visivelmente formatado.

Resultado :

000400 Este texto composto pôr um conjunto de linhas, pode 000500 ser reformatado de forma a tornar o parágrafo

000600 visivelmente formatado.

Pôr exemplo, digitando-se o comando TF36 (ou qualquer número de coluna), o

texto será reformatado para que as palavras sejam distribuídas nas linhas até a

coluna 36, isto é, a coluna 36 passa a ser a margem direita para o parágrafo em questão.

Exemplo do comando TE :

000300 Um ou mais parágrafos poderão ser incluídos

TE 400 após este parágrafo.

000500 Este parágrafo será precedido pelo texto a ser incluído

Resultado :

000300 Um ou mais parágrafos poderão ser incluídos 000400 após este parágrafo.

Em função do comando TE a tela a partir da linha 400 ficará em branco,

permitindo digitação de textos. Uma vez dado ENTER, o texto será

reformatado, isto é, texto anterior e texto incluído serão unidos, formando um

único texto.

Pôr exemplo :

000300 Um ou mais parágrafos poderão ser incluídos

000400 após este parágrafo.

000410 Este parágrafo foi incluído posteriormente pelo

000420 operador, para teste do comando TE.

000500 Este parágrafo será precedido pelo texto a ser

000600 incluído

Exemplo do comando LC :

000006 ESTA LINHA PERMANECERA COM LETRAS MAIÚSCULAS
LC3 07 ESTA LINHA E AS DUAS PRÓXIMAS, SERÃO
000008 ALTERADAS PARA QUE AS LETRAS FIQUEM
000009 EM MINÚSCULO. 1, 2,3, &, @ E ASSIM POR
000010 DIANTE, NÃO SERÃO AFETADOS.

Resultado :

000006 ESTA LINHA PERMANECERÁ COM LETRAS MAIÚSCULAS
000007 esta linha e as duas próximas, serão
000008 alteradas para que as letras fiquem
000009 em minúsculo. 1, 2, 3, &, @ e assim pôr
000010 DIANTE, NÃO SERÃO AFETADOS.

Exemplo do comando UC :

000006 esta linha permanecerá com letras minúsculas
UCC 07 esta linha e as restantes do bloco definido
000008 pelo comando uc serão alteradas para que as
UCC 09 letras fiquem em maiúsculo 1, 2, 3, &, @ e assim
000010 pôr diante, não serão afetados

Resultado :

000006 esta linha permanecerá com letras minúsculas
000007 ESTA LINHA E AS RESTANTES DO BLOCO DEFINIDO
000008 PELO COMANDO UC SERÃO ALTERADAS PARA QUE AS
000009 LETRAS FIQUEM EM MAIúSCUI-0, 1, 2, 3, &, @ E ASSIM
000010 pôr diante, não serão afetados.

2.1.6) Comandos Diversos

BNDS

(Estabelece limites), **COLS** (Colunas), **MASK** (Máscara), **TABS** (Tabulação)

BNDS => Permite ver e alterar os limites correntes.

COLS => Permite ver uma linha formatada, mostrando a identificação das colunas.

MASK => Permite ver e alterar a máscara corrente.

TABS => Permite ver e alterar a tabulação corrente.

Exemplo do comando **BNDS**:

Digitando-se o comando **BNDS**, serão exibidos os limites, isto é, as margens

esquerda e direita, como no exemplo :

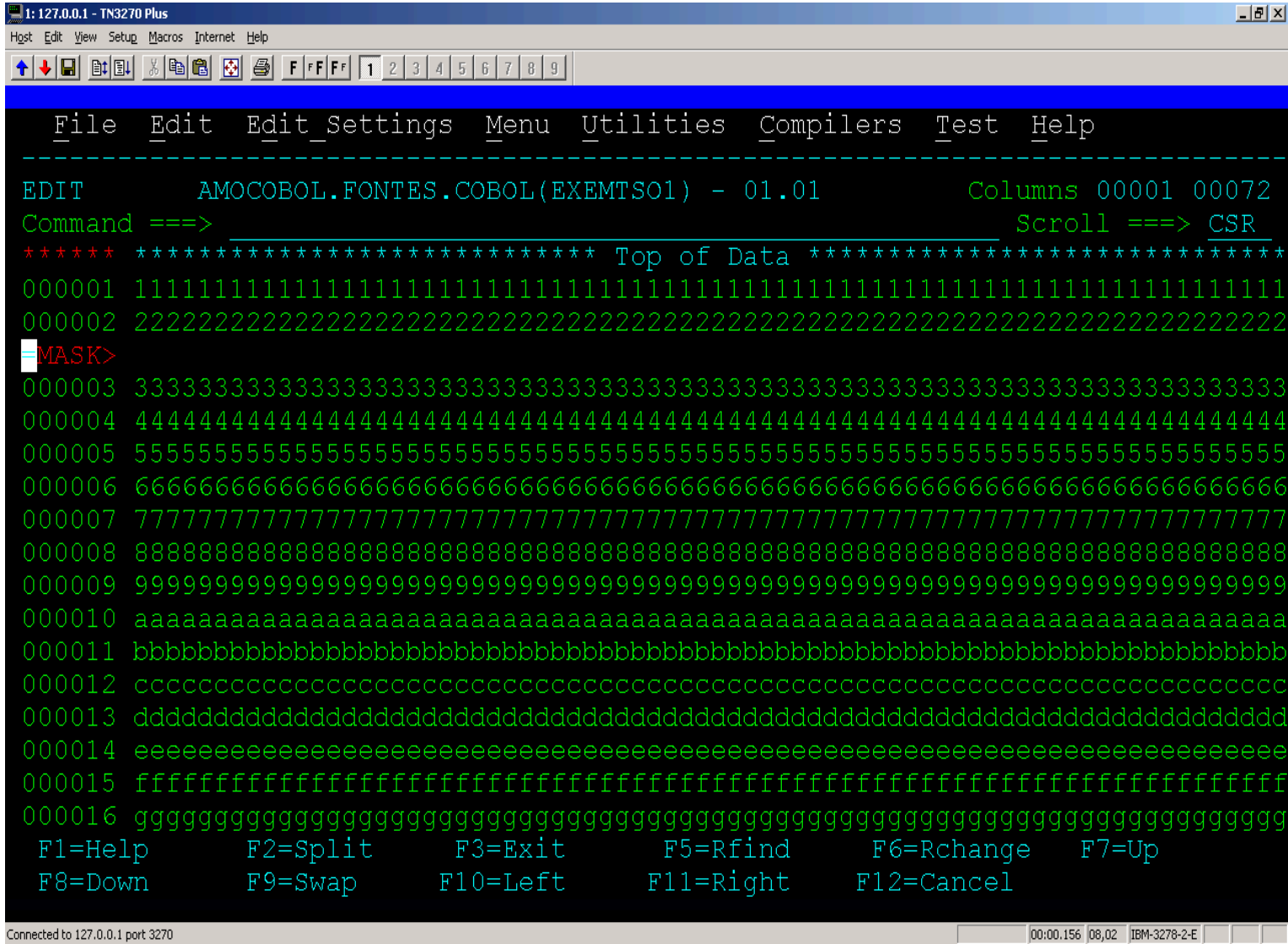
000003

=**BNDS**><>

000004

As margens poderão ser alteradas digitando-se um "<" para definir a margem esquerda, e ">" para definir a margem direita.

- Dados incluídos em modo de entrada de texto (depois de um comando TE)



Exemplo do comando **TABS** :

Digitando-se **TABS**, será exibido a linha de tabulação, Para alterar a linha, basta digitar um novo conteúdo na mesma.

Exemplo :

000001

TABS _____ * *

000002

A linha de tabulação é usada para definir campos de tabulação de hardware e de software.

“*” é usado para definir campos de tabulação de hardware

“-” ou “_” é usado para definir campos de tabulação de software.

TABULAÇÃO DE SOFTWARE => Indica o posicionamento automático do cursor após um **ENTER**. Assim, através dos símbolos “-” e “_” podemos determinar em que posições na linha, o cursor será posicionado após um **ENTER**. Pôr exemplo, o comando TAB abaixo, provocara após repetidos **ENTER** :

TABS _____ -

- Posicionamento do cursor no primeiro caractere não branco entre as colunas 2 e 31 da linha 2.

- Posicionamento do cursor na coluna 37 da linha 2

- Posicionamento do cursor no primeiro caractere não branco entre as colunas 2 e 31 da linha 3

- Posicionamento do cursor na coluna 37 da linha 3
etc.

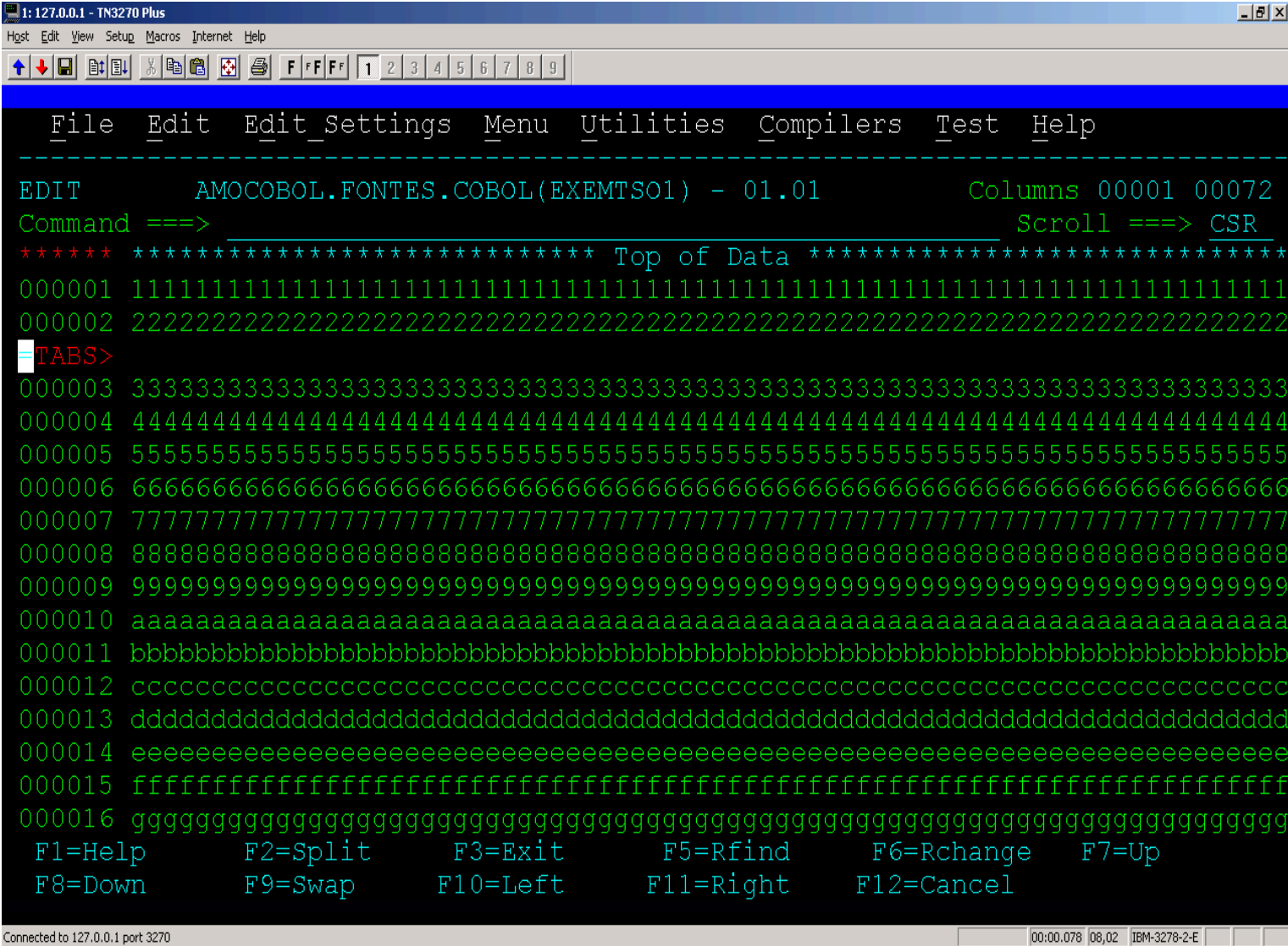
TABULAÇÃO DE HARDWARE => Permite que sejam usadas as teclas de tabulação **FORWARD**, **BACKWARD**, e **NEW-LINE**. Assim, se for definida a seguinte tabulação de hardware, via comando **TABS**, o efeito será o seguinte :

TABS * * *

col.10 col.16 col.35

Através das teclas de tabulação, poderemos pular da coluna 1 para a 11, da 11

para a 17, da 17 para a 36, sem interrupção.



3) Comandos Primários

3.1) Diversos

LOCATE

=> Para exibir uma linha particular em um arquivo ou texto.

RESET => Para “resetar”, isto é, cancelar um estado de exibição dos dados durante a edição.

SUBMIT => Para submeter um job que esteja sendo editado no momento.

PROFILE => Para exibir a profile de edição corrente, isto é, as indicações de

defaults usados na edição, para que estes defaults possam ser controlados ou alterados.

RECOVERY => Para ligar ou desligar o modo de RECOVERY

SAVE => Para salvar os dados sendo editados.

CANCEL => Para cancelar a edição sem salvar as alterações

3.2) Find/Change/Exclude

3.3) Numeração

3.4) Controle do modo de exibição

3.5) Relacionados com saída do editor

3.6) Relacionados a dados externos

3.7) Relacionados a edição de dados

B.1 – Comandos Diversos :

Comentários e exemplos sobre o comando LOCATE :

O comando LOCATE é usado para se exibir uma linha específica dos dados sendo editados. O comando pode ser abreviado pôr LOC ou L.

Exemplos :

COMMAND ==> LOC 450 Exibe a linha 450

COMMAND ==> LOC.ZFIRST Exibe a primeira linha do texto

LOC.ZFIRST : Pode ser usado para rolar a tela de modo a poder ver a primeira linha do texto.

LOC.ZLAST : Pode ser usado para rolar a tela de modo a poder ver a última linha do texto.

COMMAND ==> LOC NEXT SPECIAL Exibe a próxima linha especial (cols, prof, tabs, mask, bnds, note)

COMMAND ==> LOC PREV X .A .B Exibe a linha anterior que foi excluída e que se encontra entre o label “.A” e “.B”

COMMAND ==> LOC FIRST ERR Exibe a primeira linha que possui o flag ==ERR>

COMMAND ==> LOC LAST CHG Exibe a última linha que ontem o flag ==CHG>

COMMAND ==> LOC CMD .A .B Exibe a próxima linha que contém um comando pendente e que se encontra entre os labels “.A” e “.B”

COMMAND ==> LOC LABEL Exibe a próxima linha que contém um label

Sintaxe do comando LOCATE :

1o. Formato

LOCATE —+—número-da-linha —+———>

||

+——- label——- +

2o. Formato

LOCATE —+———+——+—X———+——+———+>

|-NEXT-| |-ERR——-| +intervalo+

|-PREV-| |-SPECIAL-|

|-FIRST-| |-ERR-----|

+ -LAST-+ |-CMD-----|

+ -LABEL-+

Comentários e exemplos do comando RESET :

O comando RESET (pode ser abreviado como RES), “limpa” a exibição do

texto sendo editado. Esta “limpeza” inclui reexibição de linhas excluídas (via comando EXCLUDE), limpeza de qualquer comando de linha pendente,

deleção de qualquer linha especial (=COLS>, =BNDS>, =MASK>, =TABS>, =NOTE>, =PROF>), e remoção de flags de linha (==CHG>, ==ERR>).

Labels também podem ser removidos via comando RESET.

Exemplos :

COMMAND ==> RESET LABEL Limpa todos os labels do texto

COMMAND ==> RESET COMMAND Deleta todos os comandos de
linha pendentes

COMMAND ==> RESET EPROR Remove todos os flags de erro

COMMAND ==> RESET CHANGE Remove todos os flags de alteração

COMMAND ==> RESET EXCLUDED Reexibe todas as linhas “escondidas”

COMMAND ==> RESET SPECIAL Deleta todas as linhas especiais

(bnds, tabs, mask, cols, prof, note)

Comentários e exemplos do comando SUBMIT :

O comando SUBMIT (pode ser abreviado como SUB), é usado para submeter
um job.

COMMAND ==> SUBMIT Submete os dados sendo editados

O comando SUBMIT é usado para submeter um ou mais jobs.

Comentários e exemplos do comando PROFILE :

O comando PROFILE (pode ser abreviado como PROF ou PRO) é usado para exibir uma ou mais linhas da profile de edição sendo usada no momento, ou para alterar alguma opção da profile.

COMMAND ==> PROFILE Exibe a profile corrente

Se for digitado um número com o comando, isto indicará o número de linhas com informações sobre a profile (0-8) que serão exibidas.

Se for digitado um nome junto com o comando, isto provocará o salvamento da profile ativa no momento, e a designação da profile especificada pelo nome como a corrente a partir daquele momento.

As informações exibidas como resultado do comando PROFILE são parecidas com :

.... COBOL (FIXED-80) RECOVERY ON NUMBER ON STD COB

.... CAPS ON HEX OFF NULLS OFF TABS OFF

.... AUTOSAVE ON AUTONUM ON AUTOLIST OFF STATS ON
PROFILE UNLOCK IMACRO NONE PACK OFF NOTEON

Comentários e exemplos sobre o comando RECOVERY :

O comando RECOVERY (pode ser abreviado como REC), liga ou desliga o modo de recuperação (recovery). Se o modo de recuperação estiver ligado quando ocorrer uma queda do sistema, será providenciada restauração do ambiente sendo usado no momento da edição e do erro.

Exemplo :

COMMAND ==> REC Liga “recovery”

COMMAND ==> REC ON Liga “recovery”

COMMAND ==> REC OFF Desliga “recovery”

Comentários e exemplos sobre o comando SAVE :

O comando SAVE força o salvamento das alterações realizadas no texto sem a necessidade de se imprimir os dados, ou de terminar a edição.

COMMAND ==> SAVE Salva o texto

O trabalho de edição pode ser retomado após o comando SAVE.

Comentários e exemplos sobre o comando CANCEL :

O comando CANCEL (pode ser abreviado como CAN), termina a edição sem salvar os dados.

COMMAND ==> CANCEL Cancela a edição do texto

Quando o comando CANCEL é usado, qualquer comando pendente ou mensagem de erro é cancelado.

B.2 – Comandos FIND / CHANGE / EXCLUDE :

FIND => Para “achar” um conjunto específico de caracteres no texto

CHANGE => Para “achar” e “alterar” um conjunto específico de caracteres no texto.

EXCLUDE => Para excluir (esconder) linhas que contém um conjunto específico de caracteres.

Comentários e exemplos sobre o comando FIND :

O comando FIND é usado para achar e exibir a próxima ocorrência de um determinado conjunto de caracteres no texto sendo editado o comando pode ser abreviado como F.

Exemplos :

COMMAND ==> FIND XXX Acha o próximo “XXX”.

COMMAND ==> FIND DEMO Acha o próximo “DEMO”.

COMMAND ==> FIND * 40 60 Acha o próximo “DEMO” localizado entre as colunas 40 a 60.

COMMAND ==> FIND ‘a = b’ Acha a próxima ocorrência dos caracteres ‘a = b’.

As aspas simples são usadas para poder se achar conjunto de caracteres que contenham brancos.

COMMAND ==> FIND “do it” Acha “do it” (que contém um branco)

COMMAND ==> FIND “*” Acha “*” (o caractere asterisco)

COMMAND ==> FIND ‘ALL’ FIRST Acha a primeira ocorrência da palavra “ALL”.

COMMAND ==> FIND ALL ‘FIRST’ Acha todas as ocorrências da palavra “FIRST”.

COMMAND ==> FIND X’00’ Acha o primeiro dígito hexadecimal com valor X’00’.

COMMAND ==> FIND ‘FFFF’X Acha o primeiro dígito hexadecimal com valor X’FFFF’.

Um caractere de picture pode ser usado para descrever um tipo de caracteres que se deseja achar.

Exemplo :

FIND P’-.’ 1 Acha um caractere não branco na coluna 1

Os caracteres listados abaixo, representam tipos de caracteres que podem ser pesquisados. Para usar esta facilidade, basta usar um dos caracteres abaixo, precedido ou seguido pela letra “P”.

= Qualquer caractere . Caracteres inválidos

@ Caracteres alfabéticos – Caracteres não numéricos

Caracteres numéricos < Carac. alfab. em minúsculo

\$ Caracteres especiais > Carac. alfab. em maiúsculo

-. Caracteres não branco

Exemplos :

FIND P’.’ 73 80 Acha caracteres inválidos entre as colunas 73 a 80

FIND P’###’ Acha números de 3 dígitos

FIND '@1'P 1 Acha labels a1, b1, c1, etc. na coluna 1

FIND P'<' Acha o próximo caractere em minúsculo

FIND P'-.' 72 Acha o próximo caractere não branco na coluna 72

Podemos definir um conjunto de caracteres seguido ou precedido pela letra

“T”, se quisermos achar este conjunto independente do fato do conjunto de

caracteres estar em maiúsculo ou minúsculo.

Exemplo :

FIND T'this' Acha o texto “this” mesmo que ele esteja em maiúsculo ou minúsculo

Para achar um conjunto de caracteres exatamente igual ao passado no comando (levando-se em conta letras maiúsculas e minúsculas), devemos digitar a letra

“C” antes ou depois do conjunto de caracteres.

Exemplo :

FIND C'This' Acha os caracteres “This” mas não “this”, “THIS”, etc.

Mais exemplos do comando FIND :

FIND XXX PREFIX X Procura conjunto de caracteres prefixados pôr “XXX” nas linhas escondidas

FIND XXX SUFFIX NX Procura conjunto de caracteres seguidos pôr “XXX”

nas linhas não escondidas

OBS.: Podemos usar o comando RFIND (PF5 ou PF17) para repetir o ultimo comando FIND usado. Pressionando-se a PF correspondente ao comando RFIND repetidamente, podemos achar ocorrências sucessivas de um conjunto de caracteres específico.

Comentários e exemplos sobre o comando CHANGE :

O comando CHANGE pode ser usado para achar e alterar a próxima

ocorrência de um determinado conjunto de caracteres em um texto sendo editado.

Exemplos :

CHANGE XXX YYY Altera o próximo “XXX” pôr “YYY”

CHANGE DEMO TESTE Altera o próximo “DEMO” pôr “TESTE”

CHANGE * * 40 60 Altera o próximo “DEMO” localizado entre as colunas

40 a 60 pôr “TESTE”.

C ‘a = b’ ‘c = d’ Altera o próximo ‘a = b’ (possui brancos)

pôr ‘c = d’ (também possui brancos)

CHG “don’t” do Altera don’t para do

CHG ““fast”” “slow”” Altera “fast” para “slow”

C ‘*’ ‘**’ Altera um único asterisco pôr asterisco duplo

CHANGE FIRST ‘ALL’ MANY Altera a primeira ocorrência da palavra ALL

pôr MANY

CHG ‘FIRST’ ‘1ST’ ALL Altera todas as ocorrências da palavra FIRST pôr 1ST

CHG ‘1’ ‘2’ 1 Altera o dígito ‘1’ para ‘2’ na coluna 1

CHG X’00’ ‘FF’X Altera o dígito hexadecimal X’00’ pôr X’FF’

CHG ALL P’-.’ ‘X’ 72 Altera todos os caracteres não brancos na coluna 72
pelo caractere ‘X’

CHANGE P’.’ X’00’ Altera o próximo caractere inválido pelo hexadecimal X’00’

CHANGE P’####’ 100 Altera o próximo número de 3 dígitos pôr 100

CHG ALL 73 80 P’=’ ” ” ” Altera qualquer caractere entre as colunas 73 a 80
pôr brancos

C ALL P’_’ 1 10 “0” Altera todos os caracteres não numéricos existentes
entre as colunas 1 a 10 pelo caractere “0”

C P’<’ P’>’ Altera qualquer letra minúscula em maiúscula

OBS.1 : Todas as outras opções comentadas no comando FIND (como uso da letra
“T”, da letra “C”, etc.) são válidas para o comando CHANGE.

OBS.2 : O comando RCHANGE (PF6 ou PF18) pode ser usado para repetir o último
comando CHANGE usado. O comando RFIND (PF5) pode

ser passado para executar a busca mas não a alteração do último comando CHANGE.

Comentários e exemplos do comando EXCLUDE :

O comando EXCLUDE é usado para excluir (esconder) linhas específicas de um texto sendo editado. Pode-se digitar o comando como : EXCLUDE, EX, ou X.

Exemplo :

COMMAND ==> EX ALL "*" 72 Exclui (esconde) linhas que contém "*" na coluna 72

As diferenças entre os comandos FIND e EXCLUDE são :

- O comando FIND altera linhas excluídas para não excluídas
- O comando EXCLUDE altera linhas não excluídas para excluídas
- O comando FIND pôr default, pesquisa todas as linhas
- O comando EXCLUDE pôr default, pesquisa somente linhas não excluídas
- O comando FIND permite a utilização dos parâmetros 'X' e 'NX'

B.3 – Comandos de numeração :

NUMBER => Liga ou desliga a facilidade de numeração de linhas (NUMBER)

RENUM => Para ligar a facilidade de numeração de linhas e para renumerar as mesmas

UNNUM => Para desligar a facilidade de numeração de linhas e eliminar a seqüência numérica

Comentários e exemplos do comando NUMBER :

Use o comando NUMBER (pode ser abreviado pôr NUM), para ligar ou desligar a facilidade de numeração de linhas e para indicar se deseja-se a numeração padrão, a numeração própria para COBOL, ou as duas (numeração padrão é o default).

Exemplos :

COMMAND ==> NUMBER Liga numeração padrão

COMMAND ==> NUMBER STD Liga numeração padrão

COMMAND ==> NUMBER COBOL Liga numeração para COBOL

COMMAND ==> NUMBER STD COBOL Liga numeração padrão e para COBOL

COMMAND ==> NUMBER OFF Desliga numeração

Comentários sobre o comando RENUM :

Use o comando RENUM (pode ser abreviado pôr REN) para renumerar as linhas do texto, iniciando-se em 100 e incrementando-se o numero da linha de 100 em 100. O comando RENUM pode ser usado também para “ligar” a facilidade de numeração de linhas. Ele aceita os mesmos parâmetros do comando NUMBER.

Comentários sobre o comando UNNUM :

Use o comando UNNUM (pode ser abreviado pôr UNN), para eliminar a seqüência numérica e desligar a facilidade de numeração de linhas. Se você quiser branquear a seqüência numérica, mas se, a facilidade de numeração de linhas estiver desligada, entre com o comando NUMBER e em seguida, entre com o comando UNNUM.

B.4 – Comandos de controle do modo de exibição :

CAPS => Usado para ligar ou desligar o modo de CAPS

IIEX => Usado para desligar o modo de exibição em hexadecimal e para especificar o formato de exibição em hexadecimal

NULLS => Usado para desligar o modo de NULLS e para especificar o tipo de nulos

TABS => Usado para desligar a tabulação e para especificar os caracteres de tabulação

NOTE => Usado para ligar ou desligar a facilidade de notas (notes)

BOUNDS => Usado para ajustar as margens esquerda e direita

Comentários e exemplos sobre o comando CAPS :

Use o comando CAPS para ligar ou desligar o modo de caps ("on" – ligado é o default). Se o modo caps estiver ligado, o editor fará com que todas as letras minúsculas sejam exibidas em maiúsculo.

Se o modo CAPS estiver desligado, os dados serão exibidos do jeito que se encontram.

COMMAND ==> CAPS Liga caps

COMMAND ==> CAPS OFF Desliga caps

Comentários e exemplos sobre o comando HEX :

Use o comando HEX para ligar ou desligar o modo de exibição em hexadecimal ("on" – ligado é o default) e para indicar quando a exibição será em formato vertical ou de dados (modo vertical é o default).

COMMAND ==> HEX Liga a exibição em hexadecimal no modo vertical

COMMAND ==> HEX VERT Liga a exibição em hexadecimal no modo vertical

COMMAND ==> HEX DATA Liga a exibição em hexadecimal no modo de dados

COMMAND ==> HEX OFF Desliga a exibição em hexadecimal

Comentários e exemplos do comando NULLS :

Use o comando NULLS para ligar ou desligar o uso de nulos ("on" – ligado é o default) e para indicar se o modo de exibição será "nulls std" ou "nulls all" ("nulls std" é o default).

No modo "nulls std", os caracteres nulos substituirão todos os caracteres brancos existentes no final de cada linha, mas se forem digitados brancos, estes serão mantidos.

No modo “nulls all”, os caracteres nulos substituirão todos os caracteres

brancos existentes no final de cada linha, inclusive brancos que tenham sido digitados propositalmente.

COMMAND ==> NULLS Liga o uso de nulos no modo padrão (“nulls std”)

COMMAND ==> NULLS ALL Liga o uso de nulos no modo ‘all’ (“nulls all”)

COMMAND ==> NULLS OFF Desliga o uso de nulos

Comentários e exemplos sobre o comando TABS :

Use o comando TABS para ligar ou desligar a facilidade de tabulação (“on” ligado é o default) e para indicar qual o caractere de tabulação em efeito (tabulação de hardware, isto é, as teclas TAB, é o default).

COMMAND ==> TABS Liga a facilidade de tabulação e usa as teclas TAB, como padrão

COMMAND ==> TABS] Liga a facilidade de tabulação e usa a tecla ‘]’ como sinal para indicar tabulação

Comentários e exemplos sobre o comando NOTE :

Use o comando NOTE para ligar ou desligar a facilidade de poder ver notas relacionadas com o texto (“on” – ligado é o default).

COMMAND ==> NOTE Notas são incluídas no texto

COMMAND ==> NOTE ON Notas são incluídas no texto

COMMAND ==> NOTE OFF Nenhuma nota será incluída no texto

Comentários e exemplos sobre o comando BOUNDS :

Use o comando BOUNDS para ajustar as margens esquerda e direita. Existe um caminho alternativo para se ajustar as margens, que é, o comando de linha

BNDS (=BNDS>). Este comando pode ser abreviado pôr BNDS ou BND.

COMMAND ==> BNDS Retorna as margens para suas posições default

COMMAND ==> BNDS 20 60 Marca a margem esquerda para a coluna 20
e a margem direita para a coluna 60

COMMAND ==> BNDS 10 * Marca a margem esquerda para a coluna 10
e deixa a margem direita como está.

COMMAND ==> BNDS * 65 Marca a margem direita para a coluna 65 e
deixa a margem esquerda como está

Deve-se especificar valores correspondentes para as duas margens, ou para nenhuma das duas. O primeiro valor passado (margem esquerda) deve ser menor que o segundo (margem direita).

B.5 – Comandos Relacionados Com O Processo De Saída Do Editor :

AUTONUM => Liga ou desliga a facilidade de AUTONUM

AUTOLIST => Liga ou desliga a facilidade de AUTOLIST

STATS => Liga ou desliga a facilidade de se manter estatísticas sobre o
modulo sendo editado

AUTOSAVE => Liga ou desliga a facilidade de AUTOSAVE

PACK => Liga ou desliga a facilidade de compactação de dados no momento
do salvamento

VERSION => Permite atribuir um número de versão para ser armazenado com
as estatísticas

LEVEL => Permite atribuir um número de nível de modificação para ser
armazenado com as estatísticas

Comentários e exemplos sobre o comando AUTONUM :

Use o comando AUTONUM para ligar ou desligar ("on" – ligado é o default)
a facilidade de numeração automática. Se tanto AUTONUM como NUMBER
estiverem ligados, os campos de seqüência numérica serão automaticamente
renumerados quando o texto for salvo. Isto pode ocorrer quando forem

executados os comandos SAVE, CREATE, REPLACE ou END. Quando o comando SAVE é acionado os dados exibidos não são renumerados, mas os dados salvos são. Se a facilidade de numeração automática estiver desligada, os campos de sequência numérica não serão renumerados quando o texto for salvo.

COMMAND ==> AUTONUM Liga AUTONUM

COMMAND ==> AUTONUM OFF Desliga AUTONUN

Comentários e exemplos sobre o comando AUTOLIST :

Use o comando AUTOLIST para ligar ou desligar ("on" – ligado é o default) a facilidade de listagem automática. Se a facilidade de listagem automática estiver ligada, os dados do texto serão automaticamente formatados e gravados

em um arquivo ISPF de impressão (list file) quando o texto for salvo.

COMMAND ==> AUTOLIST Liga AUTOLIST

COMMAND ==> AUTOLIST OFF Desliga AUTOLIST

Comentários e exemplos sobre o comando STATS :

Use o comando STATS para ligar ou desligar a facilidade de estatística ("on" –

ligado é o default). Se a facilidade de estatística estiver ligada, estatísticas se-

rão geradas para membros da biblioteca.

COMMAND ==> STATS Liga a facilidade de estatística

COMMAND ==> STATS OFF Desliga a facilidade de estatística.

A facilidade de estatísticas está sempre desligada para arquivos seqüenciais.

Comentários e exemplos sobre o comando AUTOSAVE :

Use o comando AUTOSAVE para ligar ou desligar ("on" – ligado é o default)

a facilidade de salvamento automático. Esta informação é armazenada na profile de edição.

- Se AUTOSAVE estiver ligada ("on"), o comando END (PF3) fará com que o texto seja salvo.

- Se AUTOSAVE estiver desligada ("off") com a opção PROMPT, o comando END fará com que o usuário seja questionado e levado a executar os comandos SAVE ou CANCEL.

- Se AUTOSAVE estiver desligada ("off") com a opção NOPROMPT, o comando END fará com que o texto não seja salvo, e neste caso, seu efeito será semelhante ao do comando CANCEL.

COMMAND ==> AUTOSAVE Dados atualizados serão automaticamente salvos com o comando END

COMMAND ==> AUTOSAVE ON Dados atualizados serão automaticamente salvos com o comando END

COMMAND ==> AUTOSAVE OFF O comando END fará com que o usuário tenha que digitar o comando SAVE ou o CANCEL

COMMAND ==> AUTOSAVE OFF PROMPT O comando END fará com que o usuário tenha que digitar o comando SAVE ou o CANCEL

COMMAND ==> AUTOSAVE OFF NOPROMPT O comando END ira agir como o comando CANCEL

Comentários e exemplos sobre o comando PACK :

Use o comando PACK para ligar ou desligar ("on" é o default) a facilidade de compactação. Se a facilidade de compactação estiver ligada e o texto for salvo (via comandos SAVE ou END), os dados serão armazenados em

formato compactado. Se PACK estiver desligada, os dados serão armazenados no formato padrão.

COMMAND ==> PACK Os dados serão compactados quando forem salvos

COMMAND ==> PACK OFF Os dados serão armazenados no formato padrão

Os dados serão automaticamente descompactados quando o texto for lido para edição. A facilidade de compactação tem pôr objetivo economizar espaço em disco.

Comentários e exemplos sobre o comando VERSION :

O comando VERSION é usado para alterar o nível de versão do membro

sendo editado. STATS deve estar ligada. O nível da versão do texto pode ser fornecido como um número entre 1-99. VERS ou VER pode ser usado como abreviação.

COMMAND ==> VERSION 2 Ajusta o nível da versão para “02”

COMMAND ==> VERS 12 Ajusta o nível da versão para “12”

Normalmente o nível de modificação é incrementado de um quando é realizada alguma modificação no membro.

B.6 – Comandos Relacionados A Dados Externos :

Os comandos relacionados com dados externos e listados abaixo, permitem mover ou copiar dados de uma fonte externa (arquivo seqüencial ou membro de um arquivo particionado) para o membro sendo editado; ou mover ou copiar dados do membro sendo editado para um arquivo seqüencial ou membro de um PDS.

CREATE ==> Para criar um novo membro em um PDS

REPLACE ==> Para substituir o conteúdo de um membro ou de um arquivo seqüencial

COPY ==> Para copiar dados de um membro de um PDS ou de um arquivo seqüencial

MOVE ==> Para mover (copiar e deletar) um membro de um PDS ou um arquivo seqüencial

MODEL ==> Para copiar um modelo predefinido

Os comandos CREATE, REPLACE, COPY e MOVE são similares não só no fato de poderem ser fornecidos com ou sem um nome de membro; mas também pelo fato de todos precisarem de referências a linhas. Estas referências são fornecidas através de operandos adicionais ou através de comandos de linha usados no mesmo tempo.

Nos comandos CREATE ou REPLACE, você deve especificar as linhas do texto que serão usadas.

Você pode fornecer um intervalo de linhas via operando, ou usando os comandos de linha C, CC, M, ou MM, para que estas linhas sejam colocadas em um membro externo.

Nos comandos MOVE, COPY ou MODEL, você deve especificar o destino das linhas a serem movidas ou copiadas. Pode-se usar para isto, os comandos de linha A ou B, ou as palavras chave AFTER ou BEFORE para indicar para onde os dados externos devem ir.

Digite o comando com um nome de membro se você quiser usar um membro contido no mesmo PDS do membro sendo editado.

Digite o comando sem um nome de membro se você quiser usar um membro de um PDS diferente, ou se você quiser usar um arquivo seqüencial.

Comentários e exemplos sobre o comando CREATE :

O comando CREATE (pode ser abreviado como CRE) é usado para criar um novo membro de um PDS a partir de uma ou mais linhas do texto sendo

editado.

Para especificar as linhas que serão postas no novo membro, use :

- Os comandos de linha C ou CC ou um operando corri o intervalo de linhas que

serão copiadas

- Os comandos de linha M ou MM para mover as linhas (copiar, depois deletar do fonte)

Exemplos :

COMMAND ==> CREATE MEMNAM Cria um novo membro de nome

“MEMNAM”

COMMAND ==> CRE .STRT .END Cria um novo membro que com-

terá as linhas entre os labels .STRT e .END. O usuário rece-

berá uma tela para que sejam digitadas informações como:

nome do arquivo, nome do membro, etc.

Comentários e exemplos sobre o comando REPLACE :

O comando REPLACE (pode ser abreviado pôr REPL) é usado para

substituir o conteúdo de um membro de um arquivo particionado ou de um arquivo seqüencial, usando uma ou mais linhas do texto sendo editado.

Para especificar que linhas do texto serão colocadas no membro sendo atualizado, use :

- Os comandos de linha C ou CC, ou um operando definindo o intervalo de

linhas que se quer copiar

- Os comandos de linha M ou MM, para movimentação de linhas (cópia, em

seguida, deleção)

Se o nome do membro for passado no comando, e os comandos de linha forem fornecidos, este membro terá seu conteúdo substituído imediatamente.

COMMAND ==> REPLACE MEMNAM Substitui o conteúdo do membro

“MEMNAN”

Se o comando REPLACE for fornecido sem o nome do membro, será exibido um painel, para que informações como nome do arquivo, nome do membro, etc., sejam fornecidos.

COMMAND ==> REM .STRT .END Exibe painel adicional para entrada de informações relacionadas com o destino das linhas incluídas entre os labels .STRT e .END

Comentários e exemplos sobre o comando COPY :

O comando COPY é usado para copiar uma ou mais linhas de um arquivo seqüencial ou de um membro de um PDS, para o membro sendo editado no momento.

Use os comandos de linha A (after – após) ou B (before – antes), ou as palavras AFTER ou BEFORE, após o nome de um label, para especificar onde os dados

serão copiados.

Se o nome de um membro é especificado como parte do comando, e os comandos de linha A ou B foram corretamente passados, então, o conteúdo total do membro será copiado imediatamente para o texto sendo editado.

COMMAND ==> COPY MEMNAM Copia o membro MENNAM para o texto (membro) sendo editado

Se o comando COPY for fornecido sem o nome do membro, será exibido um painel, para que informações como nome do arquivo, nome do membro, etc. sejam fornecidas.

COMMAND ==> COPY AFTER .HERE Será exibido um painel para que seja fornecido o nome do arquivo ou membro que será copiado após o label .HERE.

Você pode copiar o membro inteiro, ou uma ou mais linhas deste. Para isto,

basta fornecer o intervalo de linhas que se deseja copiar no painel que é exibido quando se emite o comando COPY sem o nome do membro.

FIRST LINE ==> 50000

LAST LINE ==> 270000

NUMBER TYPE ==> S (standard, COBOL, ou relative)

Neste exemplo, será copiado o intervalo entre as linhas 50000 e 270000 do membro fornecido no painel.

Se for fornecido “S”, a sequência numérica padrão será usada para se achar as linhas. O número de linha será checado para validação, antes da execução da

cópia.

Se for especificado “R” (relative), o número de sequência de linhas não será observado e sim a posição relativa da linha a partir do início do membro.

Pôr exemplo : se forem fornecidas as informações => 10 a 21 e “R”, então as li-

nhas entre a décima e a vigésima primeira do membro serão copiadas.

Comentários e exemplos sobre o comando MOVE :

O comando MOVE é usado para mover o membro de um PDS ou um arquivo seqüencial. O conteúdo do membro ou do arquivo é copiado para o membro sendo editado, e após isto, o membro ou arquivo origem é deletado.

Use os comandos de linha A (after – após) ou B (before – antes) ou as palavras AFTER ou BEFORE seguidas de um nome de label, para especificar onde os dados externos serão colocados no membro sendo editado.

Se um nome de membro for fornecido no comando, e a informação de destino

for passada corretamente, o membro inteiro será movido de imediato.

COMMAND ==> MOVE MEMNAM Move o membro MENINAM para

o membro sendo editado

Se o comando for emitido sem o nome do membro, será exibido um painel

adicional para que informações como nome do arquivo, nome do membro, etc.

sejam fornecidas.

COMMAND ==> MOVE BEFORE .LAB Exibe painel para se passar informações sobre a origem dos dados que serão copiados antes do label

.LAB

Comentários e exemplos sobre o comando MODEL :

Um modelo (MODEL) é composto por um grupo de instruções e notas que fornecem dados como exemplo para a criação e edição de diálogos (painéis). Modelos contêm linhas prototipadas que indicam o formato do elemento,

partes deste que podem ser alterados com dados fornecidos pela aplicação e

etc. Notas e comentários explicam a sintaxe, parâmetros, e possíveis return

codes, se necessário.

O comando MODEL é usado para associar um modelo a um arquivo sendo editado, ou para identificar a classe usada para obtenção de modelos

posteriores.

B.7 – Comandos Relacionados Com A Edição De Dados :

DELETE => Usado para deletar linhas excluídas ou não excluídas do texto sendo editado

SORT => Usado para executar funções de classificação no texto sendo editado

EDIT => Usado para causar uma entrada recursiva no editor do ISPF

Comentários e exemplos sobre o comando DELETE :

O comando DELETE é usado para deletar linhas do texto sendo editado. DEL

pode ser usado como abreviação. Um intervalo delimitando o número o de linhas

que serão afetadas pelo DELETE, pode ser fornecido.

A sintaxe do comando DELETE é :

DELETE ALL X <Intervalo>

NX

COMMAND ==> DELETE ALL X Deleta todas as linhas excluí-

das (escondidas) do texto

COMMAND ==> DELETE ALL NX .A .B Deleta todas as linhas não ex-
cluídas (exibidas na tela) existentes entre os labels .A e .B

COMMAND ==> DELETE ALL .A .B Deleta todas as linhas existen-
tes entre os labels .A e .B

COMMAND ==> DELETE X .A .B Deleta todas as linhas excluí-das existentes
entre os labels .A e .B

Os parâmetros podem ser fornecidos em qualquer ordem. A palavra ALL é
opcional quando é especificado X ou NX.

Comentários e exemplos sobre o comando SORT :

O comando SORT é usado para classificar dados.

COMMAND ==> SORT Classifica os dados em ordem ascendente.

Se você estiver usando as margens padrão e não há colunas com dados fora das
margens, então o registro inteiro será classificado.

Se você tiver alterado as margens através do comando BOUND, então somente
os dados entre as margens serão comparados e classificados. Dados fora das
margens não serão comparados e não serão movidos. Isto permite que, pôr
exemplo, se ajuste as margens nas colunas 1-36 e então se classifique os
dados

entre estas colunas, e posteriormente se ajuste as margens nas colunas 37-72
e então se classifique os dados existentes entre estas colunas.

Um ou mais parâmetros (máximo de 5) podem ser fornecidos no comando :

COMMAND ==> SORT 10 Classifica os dados entre, a coluna
10 e a margem direita

COMMAND ==> SORT 10 15 Classifica os dados entre as colunas 10 e 15

COMMAND ==> SORT 20 25 1 10 Classifica usando 2 campos de sort

A classificação pode ser em ordem ascendente ou descendente :

COMMAND ==> SORT D Classifica em ordem descendente

COMMAND ==> SORT D 10 15 Classifica dados entre as colunas 10 e 15 em ordem descendente

COMMAND ==> SORT 1 3 A 5 9 D Classifica dados entre as colunas 1 e 3 em ordem ascendente e entre as colunas 5 e 9 em ordem descendente

A classificação pode ser limitada para linhas excluídas (escondidas) ou não excluídas :

COMMAND ==> SORT NX Classifica somente linhas não excluídas

COMMAND ==> SORT X D 10 15 Classifica as linhas excluídas em ordem descendente

A classificação pode ser limitada a um intervalo de linhas :

COMMAND ==> SORT .MYLAB .ZL Classifica os dados existentes nas linhas entre os labels .MYLAB e .ZL (fim do texto).

Comentários e exemplos sobre o comando EDIT :

Emitindo o comando EDIT da linha de comandos, o editor será invocado.

COMMAND ==> EDIT ABC Faz com que o membro ABC seja editado

COMMAND ==> EDIT Faz com que o painel principal do editor

seja exibido para fornecimento de informações como :

nome do arquivo, nome do membro, etc.

O comando EDIT permite editar outro membro do mesmo arquivo, ou editar qualquer outro dado, sem encerrar a sessão corrente de edição. Quando a

sessão iniciada pelo comando EDIT for encerrada, a sessão corrente será reassumida.