

Gérson Barreiros da Silva [Fserve]

O Livro Branco do Romhacking^{1.2}

aka Tutorial De Tradução, Apostila De Tradução, Fserve's "How To" Translate Guide, White Book...

Tradu-Roms

<http://www.traduroms.cjb.net>

[1999/2004]



Tradu-Roms 2004

Fserve

1.0 Agosto, 14 de 2002

1.1 Setembro, 9 de 2002

1.1a Outubro, 18 de 2002

1.2 Junho, 12 de 2004

<http://www.traduroms.cjb.net>

* Este tutorial foi feito por Gérson Barreiros da Silva (Fserve). *
* e-mail: fserve@brasnet.org *

Este tutorial tem o intuito de apenas introduzir o leitor ao mundo do Romhacking, não tem, portanto e de forma alguma, plano de esgotar o assunto.

Copyright (C) 2004, Gérson Barreiros da Silva [Fserve]

Qualquer um que vier a reproduzir este documento, qualquer parte, precisa escrever para mim, ou pelo menos citar "Fserve/Tradu-Roms".

TUTORIAL DE ROMHACKING

Conteúdo:

- 0. PROGRAMAS NECESSÁRIOS
- 1.BASE HEXADECIMAL
- 2.MONTANDO UMA TABELA
- 3.IPS
- 4.TILES
- 5.CONVERTENDO ROMS
- 6.PONTEIROS
- 7.SCRIPT-E/SCRIPT-I
- 8.COMPRESSÕES DE DICIONÁRIO
- 9.OTIMIZAÇÃO DE SCRIPTS
- X.NOVIDADES
- Y.CONTATOS
- Z.CRÉDITOS

CAPÍTULO 0: PROGRAMAS NECESSÁRIOS [por sl0wy]

Bem, antes que comecemos a romhackear, é necessário dizer que não se traduz um jogo usando MS Word e o Paint, certo? ;)

Para facilitar nossa vida, existem vários programas prontos feitos apenas para ajudar no romhacking e na tradução de textos.

Aqui segue uma lista com os principais, e que serão utilizados no documento. À medida que precisemos deles no tutorial, entraremos em mais detalhes sobre seu funcionamento e sua utilidade.

Editores hexadecimais: <i>Direct32</i> <i>Hexpose (Hexposure)</i> <i>Hex Workshop</i> <i>Thingy</i> <i>Translhextion</i>	Editores de tiles <i>FaTiLeTy</i> <i>Naga</i> <i>TileMod2</i> <i>Tile Layer Pro</i>
Criação/aplicação de IPS <i>IPSWin</i> <i>SnesTool 1.2</i> <i>Uips</i>	Criação de tabelas: <i>Notepad</i> <i>Table Manager</i>
Assembly (65816) <i>Asm2hex</i> <i>LordTech's Snes9X</i> <i>Notepad</i>	Outros <i>IpsAndSum</i> <i>SearchRX</i>

A maioria destes programas pode ser obtida nas seguintes URLs:

www.romhacking.org <http://rpgd.emulationworld.com> www.zophar.net
<http://wakdhacks.lfx.org> <http://stealth.romhack.net>

CAPÍTULO 1: **BASE HEXADECIMAL**

Conteúdo:

1.1.O que é?

1.2.Como utilizar nas traduções

1.1.Hexadecimal é um sistema numérico muito utilizado por programadores e etc, serve para várias coisas, mas aqui só terá uma introdução do seu uso para tradução de roms... O sistema consiste de 00h a FFh 00h no sistema decimal corresponde a 1 e FFh corresponde a 255 em decimal. Não esqueça que existem outros sistemas, binário, octal, o próprio decimal... Entre outros... Em hexadecimal, os valores iniciam assim:

Hexadecimal 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11...

Decimal 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18...

1.2. A utilização vai ser constante, desde a hora de fazer a tabela até a hora em que você for brincar com os ponteiros, e se você depois quiser ir programar, pelo menos a base você já tem, veja isso... O valor do hex "0F" é 15, então "FF" será $16 * 16 - 1$, que é 255, "FFF" será $16 * 16 * 16 - 1$, que é 4095, esse "-1" é porque o nosso sistema começa em "0" e não em 1.

CAPÍTULO 2: TABELAS

Conteúdo:

- 2.1. Para que?
- 2.2. Como montar?
 - 2.2.1. Para Usuários do Windows 95/98/ME (9x)
 - 2.2.2. Para Usuários do Windows XP
- 2.3. O que é "ASCII"

2.1. Antes de falar "Para que?" Vou explicar o que é. É uma tabela de caracteres contendo o valor de cada letra/caratere, entenda assim, a ROM (Read Only Memory ou Memória Somente de Leitura), precisa de um interpretador de códigos, que diga o que significa cada 'código', logo o jeito de 'entender' o que tem na rom, é usando um editor hexadecimal, já que os bancos de memórias são divididos em hexadecimais e etc, na tabela, cada letra do alfabeto vai corresponder a um valor em hexadecimal, exemplo 01=A 02=B e etc, logo, aonde aparecer "01" na rom irá aparecer a letra "A" e onde estiver o "02", irá aparecer o "B", parece complicado, mas não é, o principal motivo da criação de uma tabela é para podermos entender a rom, botar acentos só apertando as teclas das respectivas letras, se não, teria que ser feito tudo 'manualmente', para escrever uma frase, seria hexadecimal por hexadecimal...

2.2.1. Abra o **hexpose** (também conhecido como **hexposure**), pegue uma palavra do jogo (usando o emulador...), vamos usar a palavra "Master", como exemplo. No hexpose aperte o botão "F6" e escreva somente "aster" (retiramos o M por ser maiúsculo, e as vezes o programa não acha direito...), irá aparecer uma caixa de diálogo no programa, aperte "Y" (Sim), se aparecerem várias palavras, porém sem as letras maiúsculas, saiba que você conseguiu (se não conseguiu pegue outra palavra), tente pegar o valor de uma dessas letras maiúsculas, e tente ver o código que separa uma palavra da outra. Aperte "F9" no programa, e escreva o nome da rom seguido de ".tbl", agora abra o **Table Manager**, com ele, abra a tabela, que está salva no mesmo diretório da rom, note que nessa tabela, só aparecem as letras minúsculas, use o código do espaço que você pegou, depois use o código da letra maiúscula que você pegou, se achar, o da endstring (endstring é a quebra de linha, o comando hexa que manda o texto para a linha de baixo... observe isso em frases longas no emulador...), Agora vá ao **Table Manager**, e coloque o mouse em "01", e clique com o botão direito do mouse e coloque "Maiúsculas de A-Z", faça o mesmo com as minúsculas, coloque o valor da barra de espaço, se achar, o da endstring (endstring é a quebra de linha, o comando hexa que manda o texto para a linha de baixo... Observe isso em frases longas no emulador...) e etc... Salve a tabela, abra o **hexpose**, abra a rom, se a tabela estiver com o mesmo nome da rom, e no mesmo diretório, irá abrir a tabela automaticamente. Ex: rom.bin | rom.tbl

2.2.2. Abra o **Translhex**, abra a rom no programa (esse programa trava direto...), vá ao menu e procure por "Search Relative", procure por alguma palavra na rom (que você saiba que existe), quando o programa achar, irá mostrar umas informações preciosas, algo como "A=01" "a=17" "1=??"... Agora vá ao **Table Manager**, e coloque o mouse em "01", e clique com o botão direito do mouse e coloque "Maiúsculas de A-Z", faça o mesmo com as minúsculas, coloque o valor da barra de espaço, se achar, o da endstring (endstring é a quebra de linha, o comando hexa que manda o texto para a linha de baixo... Observe isso em frases longas no emulador...) e etc... Salve a tabela, agora aconselho usar o **Thingy32**, nele, aperte "S" para fazer uma procura, para salvar, é só fechar o programa.

2.3.Quando você estiver falando com seus amigos sobre Romhacking e um deles falar que tal jogo usa o padrão de caracteres "ASC II" (lê-se ásqüi dois), pense logo, "...então a tabela desse jogo vai ter a=61", quando o jogo usa esse padrão, nem precisa ser montada uma tabela (montar uma .tbl), a não ser que queira botar acentos.

CAPÍTULO 3: **IPS**

Conteúdo:

3.1. Definição

3.2. Aplicação

3.3. Criação

3.1. IPS (International Patch System), é um arquivo que é capaz de modificar as características de uma rom, ou de qualquer tipo de arquivo. Foi inventado principalmente para auxiliar os tradutores de roms. Pois com um IPS, não é preciso fazer o download da rom traduzida, basta pegar o IPS e aplicá-lo.

3.2. Para aplicar um IPS, você necessita de um programa, neste tutorial, usarei o SnesTools como padrão. Para aplicar, abra o programa e vá em "Usar IPS", selecione o IPS e aperte <enter>, depois selecione a ROM a ser aplicado o IPS. Para facilitar, copie o SnesTools para a pasta onde estão a ROM e o IPS.

3.3. Após fazer uma tradução, você deve criar um IPS para facilitar o download dos usuários, pois o IPS é muito menor que a rom inteira. Para criar um IPS, você deve abrir o programa SnesTools, ir em "Gerar IPS", selecionar a ROM original e depois a traduzida. O IPS será criado na pasta onde se localiza o SnesTools.

CAPÍTULO 4: **TILES**

Conteúdo:

- 4.1.O que é?
- 4.2.Para que?
- 4.3.Como edita?
- 4.3.2.Como edita roms de mega drive (genesis)?
- 4.4.Acentos

4.1.Tiles são as imagens do jogo, as letras, os personagens e etc, porém nem tudo que você vai editar num editor de tiles é uma imagem do jogo, portanto não vá dar uma de doido e sair apagando tudo que vê, edite só o que você precisa, não altere a posição das imagens e etc.

4.2.Tem jogos como, Pokémon Gold, que tem "palavras" desenhadas, ou seja, para traduzir, precisa redesenhar letra por letra, não é difícil, mas chega a ser trabalhoso, e também alterando os tiles você pode botar acentos ou caracteres não existentes na rom.

4.3.Para editar basta você ter um editor de tiles (**FATILETY**, feito pela própria Tradu-Roms, pelo membro _Faster) procurar a imagem que quer editar (você vai ver um monte de coisa louca na tela, não edite isso...) lembre-se, algumas vezes, as imagens do jogo não estão no formato padrão do sistema, exemplo, para editar as fontes de Crusader of Centy, é necessário botar o programa para formato de Game Boy e, o jogo é de Mega Drive (Genesis), edite a imagem (que não vai estar arrumada), salve e vá testar o que foi feito.

4.3.2.Editando Gráficos de uma rom de Genesis, Abra a rom no **NANA** e procure os gráficos (Mas primeiro aperte "T" e depois "5"). Copie o endereço hexadecimal e abra a ROM no **HEX WORKSHOP**, vá para aquele endereço e converta para decimal (Agora não se esqueça do número decimal), Agora abra o **GEN-SMS** e aperte "E" para converter os gráficos de MEGA DRIVE para MASTER SYSTEM, Escreva o nome da ROM de GENESIS : (Apenas em formato BIN, *.bin) Ex.: sonic1.bin, escreva o nome da ROM que você deseja mandar os gráficos: Ex.: sonic1.sms, coloque o endereço inicial da ROM : (Aquele número DECIMAL) Ex.:0, coloque o endereço final da ROM : Ex.: 1000000, agora abra aquela ROM de MASTE SYSTEM no **FATILETY**, edite, quando terminar salve, agora abra novamente o Gen-SMS e aperte "I", para mandar de volta os gráficos para ROM, nome da ROM editada: Ex.: sonic1.sms, nome da rom de GENESIS em *.bin: Ex.:sonic1.bin, e o endereço (O mesmo do endereço inicial): Ex.: 0

4.4.O principal motivo de acentuar uma tradução, é para melhorar o entendimento das frases e etc, é algo fácil e que deixa a tradução esteticamente melhor, mais profissional, primeiramente, ache as fontes do jogo, caso tenha um espaço vazio após a fonte, utilize, e caso você veja algum caractere que não tenha na sua tabela, é só ir comparando com os que já tem, e você irá saber o hex do caractere que não tem na sua tabela, caso não tenha espaço vazio após a fonte do jogo, você terá que substituir por uma letra já existente e o caractere que você botar nesse lugar vai assumir o valor hexadecimal do antigo, ou seja, altere na tabela... Digamos que existe um W e você quer por um ç, faça isso no editor de tiles, depois faça isso na sua tabela, e se for usuário de Windows 9x, salve a tabela em modo DOS.

CAPÍTULO 5: **CONVERTENDO ROMS**

Conteúdo:

5.1. Para que?

5.2.1. Convertendo Mega Drive Método 1

5.2.2. Convertendo Mega Drive Método 2

5.1. Tem roms que mesmo que você procure o texto, não irá achar, e isso acontece quando as roms de Mega Drive estão em .smd...

5.2.1 Use o **UCON** para converter arquivos nesse método. Os formatos normais de roms de Mega Drive são *.smd , mas o rom estando neste formato, torna difícil a tradução. Por isso, é necessário convertê-lo para *.bin. Pois neste formato já fica melhor. Para converter digite o seguinte (é necessário que o programa Ucon e a rom estejam no mesmo diretório):

Ucon z rom.smd /sg

5.2.2 Copie a rom **.SMD** e o programa **SMD**, tudo para uma mesma pasta, aí vá ao prompt do MS-DOS e digite:

cd pasta (obs. aonde tem "pasta" é a pasta que está o programa e a rom)

smd /CB nomedarom.smd

CAPÍTULO 6: PONTEIROS

Conteúdo:

6.1.O que é?

6.2.Como edita?

6.3.Observações

6.4.Motivos Para Mexer Em Ponteiros!

6.1.É um valor hexadecimal que aponta para alguma coisa, início do texto, imagens e etc.

6.2.Vamos pegar uma rom de exemplo, uma rom de Super Nintendo, que o texto comece no offset '12E0FF' o texto é 'my dad', como achar o ponteiro? Já falei o básico, agora é simples, basta pegar o 12E0FF subtrair pelo header (200h) do Super Nintendo e você irá achar 12DEFF, ainda não achou os ponteiros... Para achar, basta pegar o valor encontrado e tirar o par de hexadecimal da esquerda de forma que fique apenas 2 pares de hexadecimal, DE FF, agora os inverta, FF DE, pronto, FFDE é o ponteiro que aponta para o começo desse texto, e agora que você tem o valor, é só procurar na rom o hexadecimal "FFDE" mais próximo do texto, quando você tiver suspeita que achou, de uma mexidinha (escreva FFDF), se for ele mesmo, na rom vai mudar alguma coisa... Se não era ele, a rom vai funcionar como se nada tivesse acontecido ou irá travar. Agora vou explicar como funciona, quando eu passei o cursor em cima do 'm' de 'my dad', eu olhei numa parte escrita 'offset' e vi escrito 12E0FF. Pronto, agora eu peguei esse valor 12E0FF, e retirei 200h, (por que para acharmos ponteiros temos que pegar o offset e subtrair pelo valor do header do videogame, do Super Nintendo é 200h, do Nintendo é 10h), quando fiz 12E0FF - 200 achei 12DEFF, mesmo quando achei o 12DEFF, tirei os números da esquerda, deixando somente 1 par, (pois é sempre assim que estão os ponteiros), ficou DEFF, e depois disso inverti, ficando FF DE. Isso é o ponteiro, quando você achar 1, em seguida, você achou a tabela de ponteiros...

6.3.O header do Super Nintendo é 200h, do Nintendo 10h, mexer com ponteiros pode ser complicado, mas não é, se você tentar, vai pegar a manha, e vai ficar fácil, você logo no começo pode usar a calculadora do Windows em modo científico para HEXADECIMAL, depois, com o costume irá fazer os cálculos de cabeça...

6.4.Há vários motivos para usar os ponteiros, o mais comum, é a famosa falta de espaço para escrever o seu texto em português, mas há tradutores que usam ponteiros para mudar as fontes de lugar, ou então mudar um tile inteiro. Há outros (R_Lopes) que usam ponteiros para 'expandir os menus dos jogos' ou seja, ponteiro é algo muito útil para você saber.

CAPÍTULO 7: **SCRIPT-E/SCRIPT-I**

Conteúdo:

7.1.O que é?

7.2.Como Usar

7.1.Os programas, script-e e script-i (extract/insert), são programas que servem para extrair/inserir o texto de um jogo, ele extrai para um bloco de notas (ou insere de um), e é no bloco de notas que você traduz usando esses programas, você pode pedir ajuda para alguém que saiba inglês, mas não sabe romhacking.

7.2.Para usar, basta já ter um tabela completa (endstring, mte, dte, ...), e os endereços de aonde inicia o texto da rom, o programa pede a tabela, depois ele vai pedindo outras informações a respeito da rom, nome da rom, nome do .txt e etc, quando for extrair, o programa vai pedir o número de bytes a extrair, lembre-se que cada '1' letra corresponde a 1 byte.

CAPÍTULO 8: COMPRESSÕES DE DICIONÁRIO

Conteúdo:

8.1.DTE

8.2.MTE

8.3.Substring

8.4.Editando as compressões de dicionário [DTE/MTE/SUBSTRING]

8.5.Tutorial de DTE/MTE por sl0wy.

8.1.DTE, Double Tile Encoding, tem gente que ao saber que o jogo tem algum tipo de compressão, mesmo que seja simples, já para logo a tradução, não deviam fazer isso, já que as DTE's vão é ajudar... Veja como ficaria uma tabela sem DTE.

01=A

02=B

03=C

e agora veja uma tabela com dte

01=of

02=th

03=is

Caso não tenha percebido a diferença, olhe de novo até perceber que normalmente, só tem um caractere por hex, mas quando se usa DTE, tem 2, e se for MTE, terá mais...

8.2.MTE, Multiple Tile Encoding, é um sistema parecido com DTE, a diferença é que invés de apenas dois caracteres por hex, nesse cabem até frases, mas a maior utilidade é usar as palavras que são mais utilizadas, essa técnica é comumente chamada de "Otimização de Script".

8.3.Substring é outro sistema parecido com DTE, esse usa apenas parte da palavra, exemplo: "Maravilhosa" na substring poderia ser "ravilh" isso serve muito nas traduções por que tem palavras que se usa o mesmo radical, exemplo "acontece", se a substring for "acontec", digamos que tenha um valor D0 01h, então quando formos usar, é só escrever o hex D0 01, que já vai aparecer no jogo a palavra acontec, e logo depois do D0 01, escrevemos "ia", então vai formar "**acontecia**".

(8.4.Editar essas compressões simples é realmente simples :), aqui vai o método que eu uso, e nunca me deixou na mão. [serve para dte/mte/substring]

I - Alguns textos aparecem junto com DTE. Como? Vou explicar. No caso se um DTE tem a palavra "other", e você estiver procurando a palavra "Another", com bastante atenção você irá perceber que se você não achar a palavra "Another", procure então: An + DTE (other)

II - No hexpose se tiver que localizar maiúsculas e minúsculas, ele não acha. Tem de colocar um dos dois. E se você quiser localizar "Another". Coloque pra localizar: "nother", pois o "A" é maiúsculo. Se não achar coloque pra localizar: n +DTE (other)

8.5.Tutorial feito pelo sl0wy [adaptado por Fserve]

E então, o que são DTEs/MTEs/Substrings?

Elas na verdade são umas tipas muito simples de compressão chamadas compressões de dicionário, usado pelos programadores dos jogos para economizar espaço no jogo,

para isso, faz-se um verdadeiro dicionário com as palavras mais usadas no jogo, e armazena-se este dicionário em algum lugar. Assim, quando determinado byte aparecer no meio do texto, o jogo interpretará esse byte como sendo determinada palavra do dicionário, desde agora é bom esclarecer uma coisa: DTEs e MTEs são diferentes. DTE significa Dual Tile Encodig, ou seja, são apenas dois tiles (duas letras, números, qualquer coisa, contanto que sejam dois). Então caso o que está comprimido sejam três ou mais caracteres, esta seqüência de caracteres é uma MTE, ou Multiple Tile Encoding. E se for apenas o meio da palavra com uns três caracteres ou mais, vai passar a se chamar SUBSTRING.

Agora vamos analisar um exemplo:

Tenha cuidado, bravo guerreiro. Há muitos perigos nesta estrada.

Com uma compressão de dicionário, o programador poderia reduzir e muito o tamanho desta frase. Então agora suponhamos a mesma frase, agora com uma compressão de dicionário em algumas palavras, onde '*' é uma palavra comprimida (lembrando que a palavra pode ser qualquer uma, certo?):

* cuidado, bravo *. * muitos perigos nesta *.

Viu, com esta simples compressão, o programador pôde economizar muitos caracteres.

Obs: Como você pode ter visto, neste caso as palavras comprimidas são MTEs, pois são mais de 2 caracteres.

Estas palavras geralmente estão em uma espécie de dicionário com as palavras do dicionário. No caso do nosso exemplo, após você ter feito sua **tbl** [Sigla de Tabela] básica com números, letras e acentos, abra seu editor hexadecimal preferido, vá até o lugar onde está o texto (no nosso exemplo, 'Tenha cuidado bravo...'). No nosso exemplo, o editor hexadecimal mostraria algo como isso:

cuidado, bravo #. # muitos perigos nesta #.

Antes de tudo, anote os hexadecimais que estão comprimidos. Suponhamos que no nosso exemplo eles sejam DD, E3, E5 e E8. Então agora faça uma busca por estas palavras comprimidas (não pelo hexadecimal, pela palavra). No nosso exemplo eu faria uma busca pela palavra 'Tenha'. Então o editor hexadecimal me levaria para um lugar com várias palavras separadas por um byte, geralmente o byte de endstring. O editor hexadecimal então mostraria algo mais ou menos assim (onde # é o byte separador):

blabla#bleheh#Tenha#hsuhdu#aaaaaaa#jsks#
guerreiro#sadfdsf#Há#dfjss#dsfhksdj#estrada

E assim até o fim dos valores do dicionário. Agora, abra sua tbl, e adicione os valores do dicionário. No nosso exemplo nossa tbl ficaria assim:

DD=Tenha
E3=guerreiro
E5=Há
E8=estrada

Caso você queira, aproveite para adicionar os outros valores do dicionário. Nossa tbl ficaria assim:

DC=bleheh

DD=Tenha
DE=hsuhdu
DF=aaaaaaa
E0=jsks
E3=guerreiro
E5=Há
E8=estrada

E assim vai. Como você pode ver, eu coloquei apenas algumas palavras para ilustrar o funcionamento da coisa. No seu caso, você deve adicionar todas os valores, para assim ver todo o texto do jogo.

Explicação adicional: Às vezes, a MTE pode usar mais de um byte para sinalizar o valor do dicionário. Isso é usado quando o jogo tem uma grande quantidade de palavras para serem codificadas. Então geralmente são usados 2 bytes. Assim, nosso exemplo poderia ficar assim:

F0DD=Tenha
F0DE=hsuhdu

Então os 2 primeiros bytes não são alterados, sendo incrementados somente os 2 últimos bytes, como você viu acima. Então, quando os 2 últimos bytes atingem FF, os 2 primeiros bytes serão incrementados, ficando F100, no nosso exemplo. Então, quando são usados 2 bytes para se representar um valor do dicionário, estamos lidando com MTEs 16 bits, com esse nome por motivos óbvios. Também por motivos óbvios, esta coisa só acontece quando trabalhamos com MTEs, já que se usássemos 2 ou mais bytes para representar uma DTE não estaríamos economizando espaço nenhum. Agora é simples. Traduza esses valores do dicionário, e depois altere os valores da tbl. Pronto! Você achou e traduziu suas DTEs/MTEs!

CAPÍTULO 9: OTIMIZAÇÃO DE SCRIPTS [por slowy]

Antes de tudo, considere que o jogo que você está traduzindo utiliza DTEs ou MTEs. E elas servem justamente para economizar espaço, substituindo palavras ou pedaços das palavras mais usadas no texto por um número menor de bytes. Depois de traduzidos os scripts, geralmente as palavras mais utilizadas são outras. Não seria mais útil, ao invés de simplesmente traduzir a tabela de MTEs/DTEs, colocar as palavras mais utilizadas pelo seu texto traduzido na tabela? É aí que entra a chamada **otimização de scripts**: você cria um programa (ou usa algum pronto) que checa a ocorrência de strings no texto e escolhe as que mais aparecem. Um exemplo de programa que faz isso é o DTESat, um programa brasileiro encontrado em **www.brgames.org**. De qualquer forma, o ideal é a criação de um programa específico para cada tradução. O algoritmo do programa tem de equilibrar velocidade com as melhores buscas possíveis, não só pegando palavra por palavra, mas juntando trechos de palavras, até formar strings de grande ocorrência. Deixo todas essas idéias mirabolantes para quem tiver tempo e paciência para implementar. Se alguém o fizer, me avisem.

Para deixar um pouco mais claro o conceito, vamos a um exemplo.

Imaginem que esta seja a tabela de MTEs da ROM original:

back:chemical:physics:history:geography:maths

Considerem os dois pontos como endstrings.

Agora vejam um texto não traduzido:

Dragons are bad. Dragons are ugly.

Reparem que nenhuma entrada da tabela de MTEs é usada!

Agora vejam o texto traduzido:

Dragões são maus. Dragões são feios.

Por que não trocar as palavras da tabela ao invés de apenas traduzir?

Um otimizador de scripts poderia ordenar a ocorrência das palavras assim:

'Dragões ': 2 ocorrências
'são ': 2 ocorrências
'maus.': 1 ocorrência
'feios.': 1 ocorrência

Ou ainda:

'Dragões ': 2 ocorrências
'são ': 2 ocorrências
's.': 2 ocorrências
'mau': 1 ocorrência
'feio': 1 ocorrência

Depois é só substituir (automaticamente ou não) as entradas da sua tabela de MTES, que agora deveria ficar assim:

Dragões :são :s.:history:geography:maths

Nada complicado, não? Só tome aquele velho cuidado de não ultrapassar o espaço da tabela de MTEs.

CAPÍTULO X: **NOVIDADES**

1.1a -> 1.2

1. Erros gramaticais x_x

2. Botei umas figuras do zero =D

3. Renomeado, titulo sugerido pelo sI0wy

"<sI0wy> se o negocio é sobre tradução o titulo tinha que ser em português"

1.1 -> 1.1a

1.O sI0wy me mandou um capitulo inteiro, que no caso é o capitulo dos programas que devem ser utilizados...

2.Corrigi uns erros na parte de Bases Hexadecimais

3.Mais erros gramaticais corrigidos :)

1.0 -> 1.1

1.Apostila foi nomeada como 'The White Book Of The Romhacker'.

2.Adicionado uma seção de novidades. O_o

3.Botei o nick do _Faster, pow, tinha esquecido ! E agora botei nick e nome para as pessoas que conheço.

4.Uma seção de contatos... Ficou faltando e talz...

5.Corrigi milhões de erros gramaticais...

6.As ultimas seções são XYZ, por motivos óbvios (a cada seção que eu colocava, tinha que sair mudando :P)

CAPÍTULO Y: **CONTATOS**

Nome: Gérson Barreiros da Silva [Manaus / AM – BRASIL]

Idade: 16 anos [traduzo desde os 12, programo em quick basic desde os 10, tenho base de delphi, assembly, pascal, c, entre outros...]

UIN (ICQ): 42269897

MSN Messenger: fserve@hotmail.com

AOL: fservemalzao

mIRC: Fserve nos canais: #Tradu-Roms #Hexagon #Emulandia #Emuroms na rede BRASnet, irc.brasnet.org

E-Mail: fserve@brasnet.org

Home Page: <http://www.traduroms.cjb.net>



CAPÍTULO Z: CRÉDITOS

sLowy, retirei uma parte de seu tutorial de compressão de dicionário... E ele me ajudou com outras coisitas mais aki :P [se não me engano, esse cara é o Raphael Kupo, que nome hein...]
{FBI}, criador da Tradu-Roms, me ajudou nas traduções, e deixou a Tradu-Roms em minhas mãos há muito tempo... [não lembro seu nome, sei que ele é de Vitória da Conquista/BH]
Bruno Costa [Salvador/BH], membro da Tradu-Roms e criador do FATILETY aka _Faster

Agradecimentos:

Slime [porkinho da #hexagon também conhecido como groxo...], aka Alan...
Thevilman [Fernando]
Wanzeler_[Filho de satan!]
R_Lopes [Raiffy Lopes, um dos melhores tradutores que eu conheço :) ele é do Espírito Santo!]
UiS [mano velho], aka Hudson, e os outros da família...
Frankz [Francisco], Cybergoku [Raoni], Lamec [Igor] Leo_Luz [Leonardo Luz de Laguna/SC], Greengoblin- [Ian Guedes], amigos...
Meu PC [namoro ele o dia inteiro...]
A você que está lendo ;)
Toda a comunidade Romhacker!
<http://www.traduroms.cjb.net> [1999/2004]... Qualquer dia vou atualizar esse tutorial, e por coisa útil nele, falar sobre RLE, LZW, LZ77, HUFFMAN e VWF, quem sabe uma próxima versão?
Lembrando que todos os programas citados aqui, você pode pegar na home page da **Tradu-Roms**, e se tiver mais dúvidas, entre no canal da Tradu-Roms [#Tradu-Roms na BRASNET, [irc.brasnet.org](irc:brasnet.org)]
Bem... Como eu posso dizer?
See Ya!



Fserve
Tradu-Roms
[1999/2004]
<http://www.traduroms.cjb.net>
traduroms@brasnet.org
#Tradu-Roms
[BRASNET – [irc.brasnet.org](irc:brasnet.org)]