Extraindo o RGSSAD com MakeFile decompile/compile

Primeiro agradecer o dono do código FRANTJC

Segue o repositório original: https://github.com/frantjc/pokemon-debug.git

Segue repositório do meu fork: https://github.com/Brunobnhtr3/pokemon\_debug\_atualizado.git

Para usar o pokemon-debug e modiﬁcar os arquivos do Pokémon siga esse passo a passo:

* Atenção, siga o passo a passo rigorosamente caso encontre dificuldades ou erros, consulte sempre o chatgpt para orientação.

1. Pré-requisitos

* Python3 instalado

Link: <https://www.python.org/downloads/>

* Chocolatey

### Abra o PowerShell com privilégios de Administrador.

1. Se ainda não tem o Chocolatey, instale com este comando:

powershell

Set-ExecutionPolicy Bypass -Scope Process -Force; iex ((New-Object System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1'))

1. Depois execute:

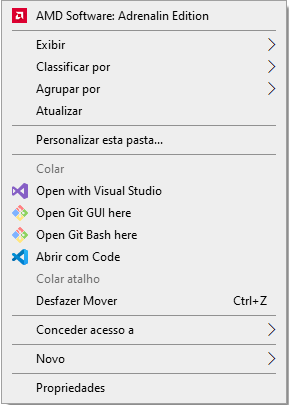
powershell

choco install make -y

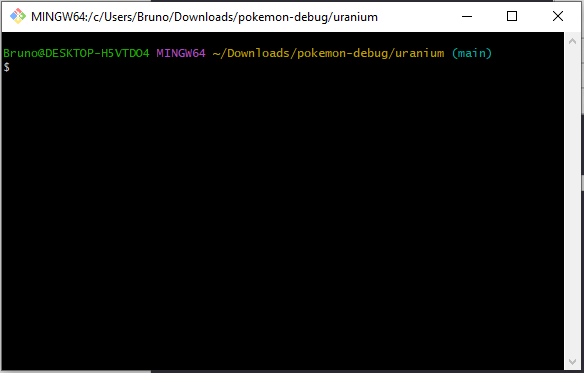
Feche e abra de novo o terminal. Agora **make --version** deve funcionar.

1. Instalar o Git para obter o git bash here  
   link: <https://git-scm.com/downloads>

Faça o download para o sistema operacional que você usa e só siga a instalação se tiver algo como “add to path”, selecione isto apenas.

 Após a instalação vá para alguma pasta onde deseja clonar o repositório do pokemon-debug , “eu escolhi a pasta Download” e clique com o botão direito do mouse e vai aparecer a opção “Open Git Bash here”

Após isto irá aparecer 1 terminal do git como abaixo



Em seguida vocês irão executar o código abaixo, para clonar o repositorio

Git clone --recursive <https://github.com/frantjc/pokemon-debug.git>

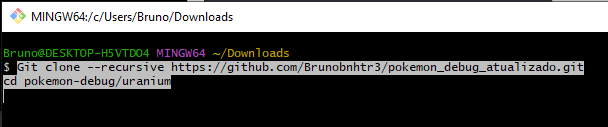
cd pokemon-debug/uranium

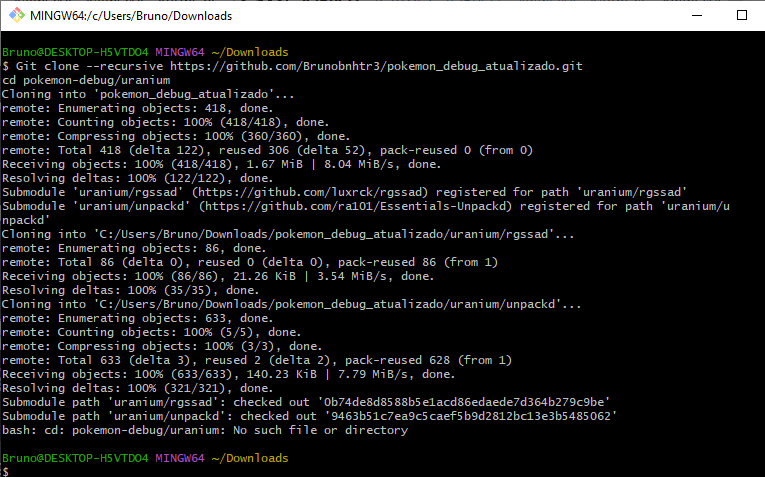
ou

Git clone --recursive https://github.com/Brunobnhtr3/pokemon\_debug\_atualizado.git

cd pokemon\_debug\_atualizado/uranium

Eu vou modificar o Uranium por isso estou direcionando para a pasta do Uranium.



Se tudo ocorrer bem vai aparecer algo como isto:  


Confirme se o submódulo está no lugar .

Vá para a pasta rgssad em pokemon\_debug\_atualizado\uranium\rgssad

Ou

Vá para a pasta rgssad em pokemon\_debug\uranium\rgssad

O diretório rgssad precisa conter um Cargo.toml. Se ele não estiver lá, o Makefile não vai conseguir rodar o processo de decompilação.   
  
Se não exister Execute, na pasta uranium, se existir pule este passo!

git submodule update --init --recursive

Isso baixa qualquer parte interna (se houver) que o repo precise.

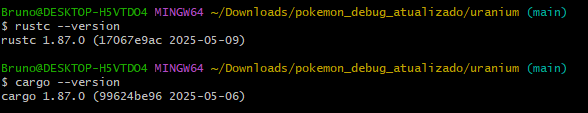
1. Instale o Rust e o CARGO

Vá para o link: <https://rustup.rs>  
e baixe o instalador ao abrir escolha a opção numero 1 a recomendada que precisa do visual studios para funcionar e depois é só seguir os passos.

Após isto você pode fechar e abrir o “git bash here” novamente e teste os códigos abaixo:

cargo --version

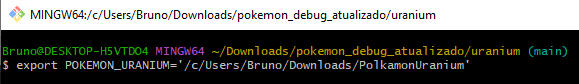
rustc --version



1. Configure a variável de ambiente No Git-Bash, defina essa variável, mas use caminhos no estilo Unix (C: /c/), remova espaços da pasta do jogo que irá ser decompilado:

No Git-Bash, defina o caminho correto:

export POKEMON\_URANIUM='/c/Users/Bruno/Downloads/PolkamonUranium'



Execute o processo de decompilação:

Ainda no Git-Bash, na raiz do repositório pokemon-debug/uranium, execute:

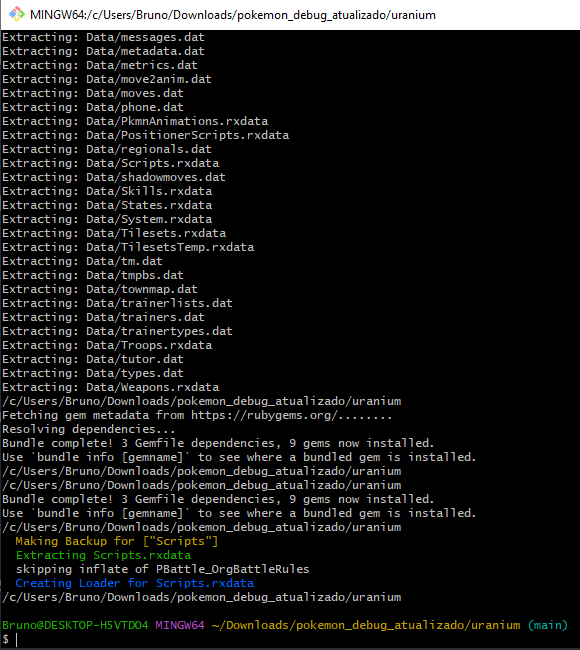
make decompile

Após isto deve ter extraído todos os arquivos do rgssad, vai ocorrer um error do tipo

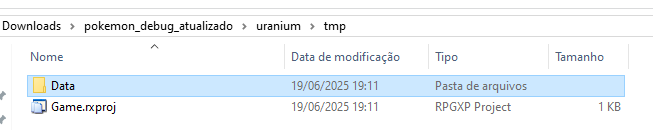
thread 'main' panicked at src\main.rs:451:13:

assertion failed: args.len() > 3 && args.len() < 6

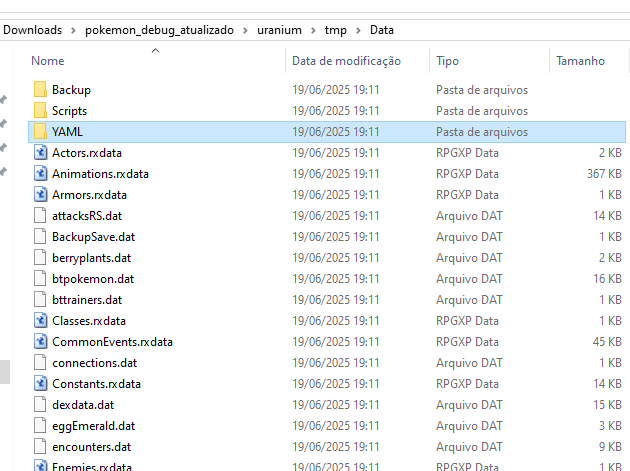
não consegui resolver este problema, mas o arquivo foi desencriptado na pasta: pokemon-debug\uranium\tmp



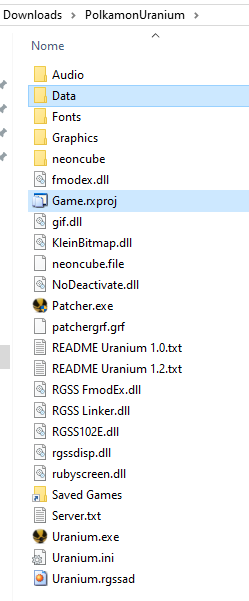
Mas ainda não terminou os arquivos scripts que foram extraídos vieram com erros , não consegui identificar o motivo, na pasta tmp temos o data.rxproj e a pasta data.

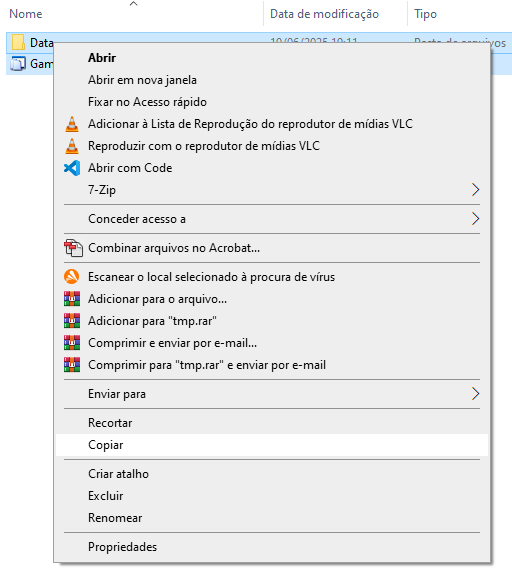
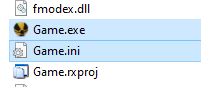
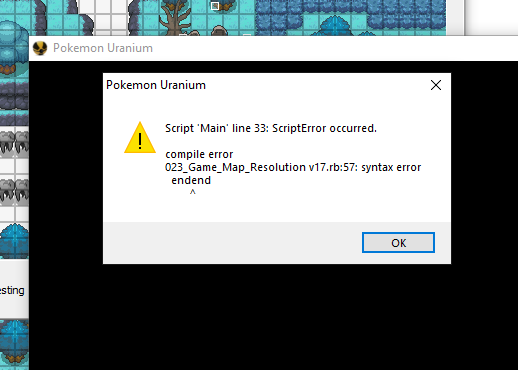


E dentro dele nos temos as pastas Backup, Scripts, YAML e os arquivos .rxdata

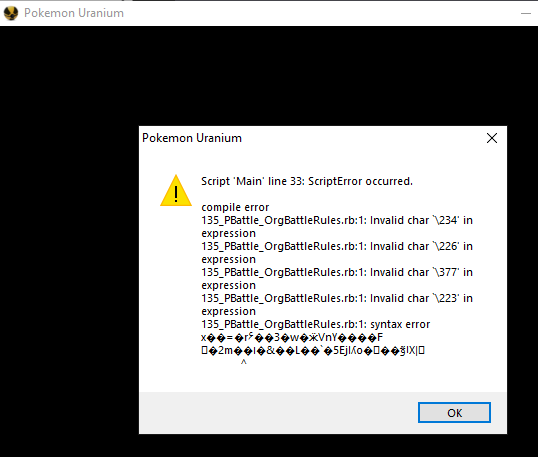


Para fazer um teste no pokemon uranium que possui seu .exe e seu .ini modificados vamos copiar a pasta data + o game.rxproj e colar na pasta do jogo. Após isto alterar o nome do Uranium.exe e Uranium.ini para Game.



Após isto abra o Game.rxproj e execute o game.

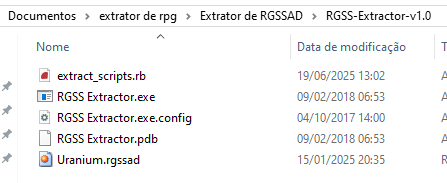
Como aparece no erro, o make decompile não decompilou certos alguns scripts do arquivo Scripts.rxdata.

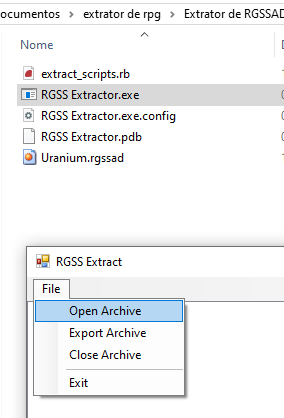
Mas você pode estar se perguntando porque não conferir o arquivo 023\_game\_map\_resolution e resolver o erro?   
R: Eu fiz isso resolvi este problema porém apareceram vários outros como este:

Então vamos precisar extrair os rxdata por outro lugar então vamos usar o RGSS EXTRACTOR, disponível também no Git do pokemon\_debug\_atualizado

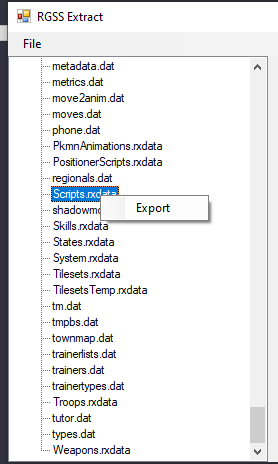
Depois de extrair o extrator você terá 3 arquivos, agora você terá que pegar o arquivo rgssad do pokemon e colar na mesma pasta do extrator e coloque também o arquivo “extract\_scripts.rb”, veio junto com o git do pokemon\_debug\_atualizado, precisa ter o Ruby instalado se não tiver baixe aqui: <https://rubyinstaller.org/>

Após isto a pasta deverá se parecer com isto:

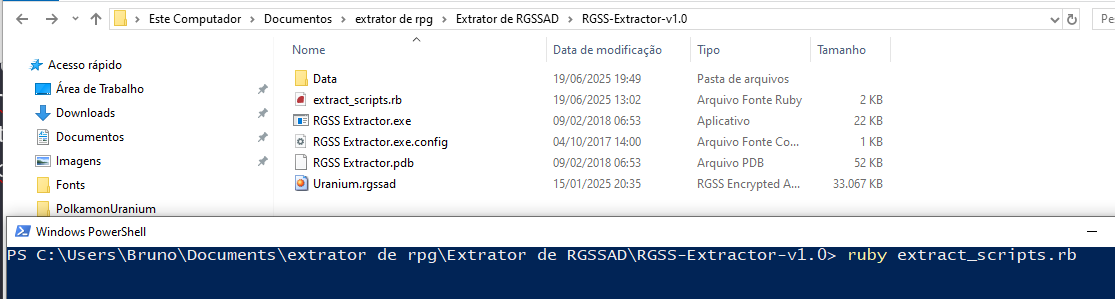


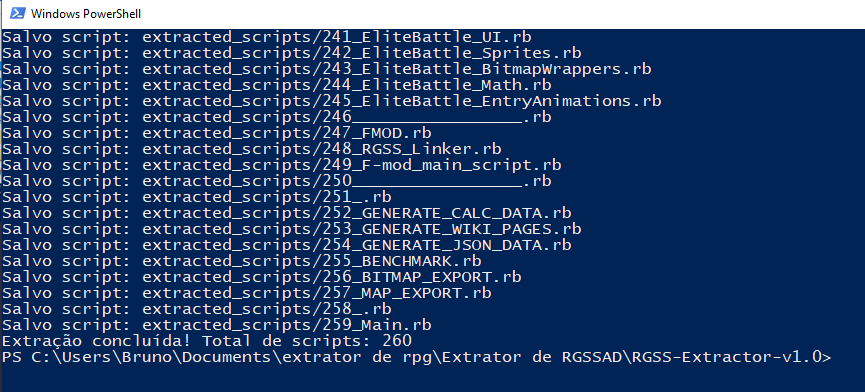
Após isto execute o arquivo RGSS Extractor.exe e abra o arquivo Uranium.rgssad

Após abrir o arquivo Uranium.rgssad, vai aparecer uma pasta chama Data clique no + e selecione o Scripts.rxdata clique com o botão direito do mouse em cima e depois Export.



Agora na pasta do extrator vai aparecer uma pasta chamada Data e dentro o nosso script.rxdata, agora vamos abrir o terminal cmd ou powershell na mesma pasta do extrator e executar o comando:  
ruby extract\_scripts.rb

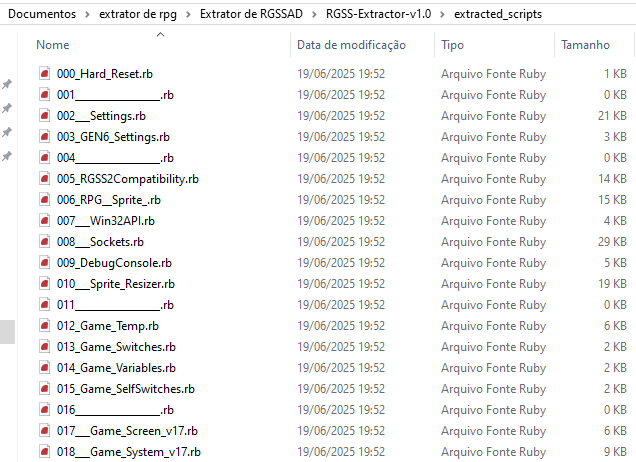


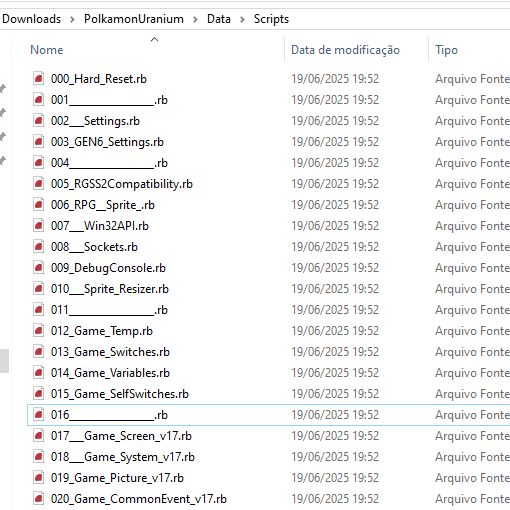
Após isto deverá aparecer algo como isto:

1 -Agora na pasta do extrator terá uma nova pasta chamada “extracted\_scripts”, abra ela e veja os scripts do rxdata extraídos.

2- Agora vamos selecionar todos esses scripts.rb e colocar na nossa pasta de scripts do jogo.

1)



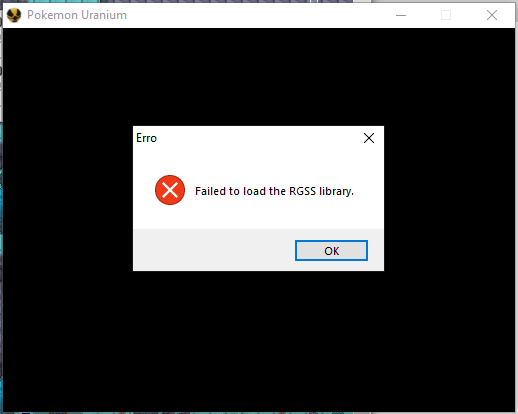
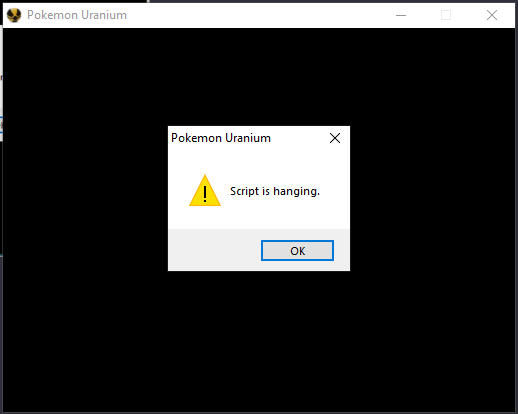
2)

Agora precisa remover ou excluir o arquivo Uranium.rgssad da pasta do jogo.

\*ATENÇÃO\*

\* Abra o rxproj e tente rodar o jogo se der algum erro de arquivo da pasta PBS, feche o rxproj

Uranium.exe e Uranium.ini e tente abrir o jogo novamente vai dar erro feche e renomeie para Game.exe e Game.ini até aparecer um erro igual este:

Não entendi o porque fica dando o erro do PBS, mas fique fazendo isso altere o nome dos .exe .ini , se possível crie um novo projeto no rpg maker xp em uma pasta aleatória copie os Game.exe,Game.ini e o rxproj, para a pasta do jogo, até obtermos o erro Failed to load the RGSS library.

Este erro se dá pelo motivo de que nosso Game.exe e Game.ini estão com os nomes incorretos para o jogo deveria estar com Uranium, então devemos alterar dentro de alguns scripts.rb estas chamadas.

O erro esta presente nos arquivos:  
\* 248\_RGSS\_Linker.rb  
\* 051\_FileTests\_v17.rb

Abra os arquivos com notepad ++ ou outro editor de sua escolha.

Precisamos renomear o trecho do código que chama o Uranium 🡪 Game





Depois de alterar e salvar os arquivos abra o jogo novamente!

