**Comparador de Programas Monolíticos**

**Êndril Castilho da Silveira, Leonardo Pellegrini Silva**

Disciplina de Teoria da Computação

Departamento de Informática – UNISC

Campus de Santa Cruz do Sul

96.810-206 - Santa Cruz do Sul - RS - Brasil

[endrilcs@mx2.unisc.br](mailto:endrilcs@mx2.unisc.br), leonardopellegrini@mx2.unisc.br

**1. Sobre o projeto**

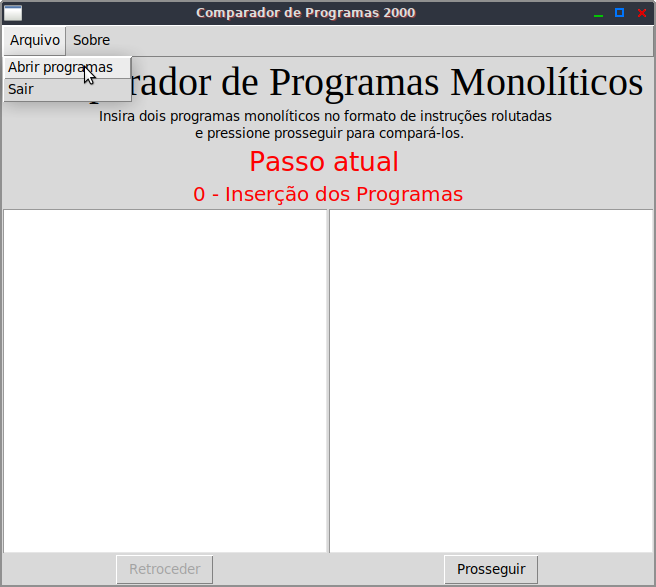
Desenvolvido em Python 3, o aplicativo conta com uma interface gráfica construída com base na ferramenta Tkinter.

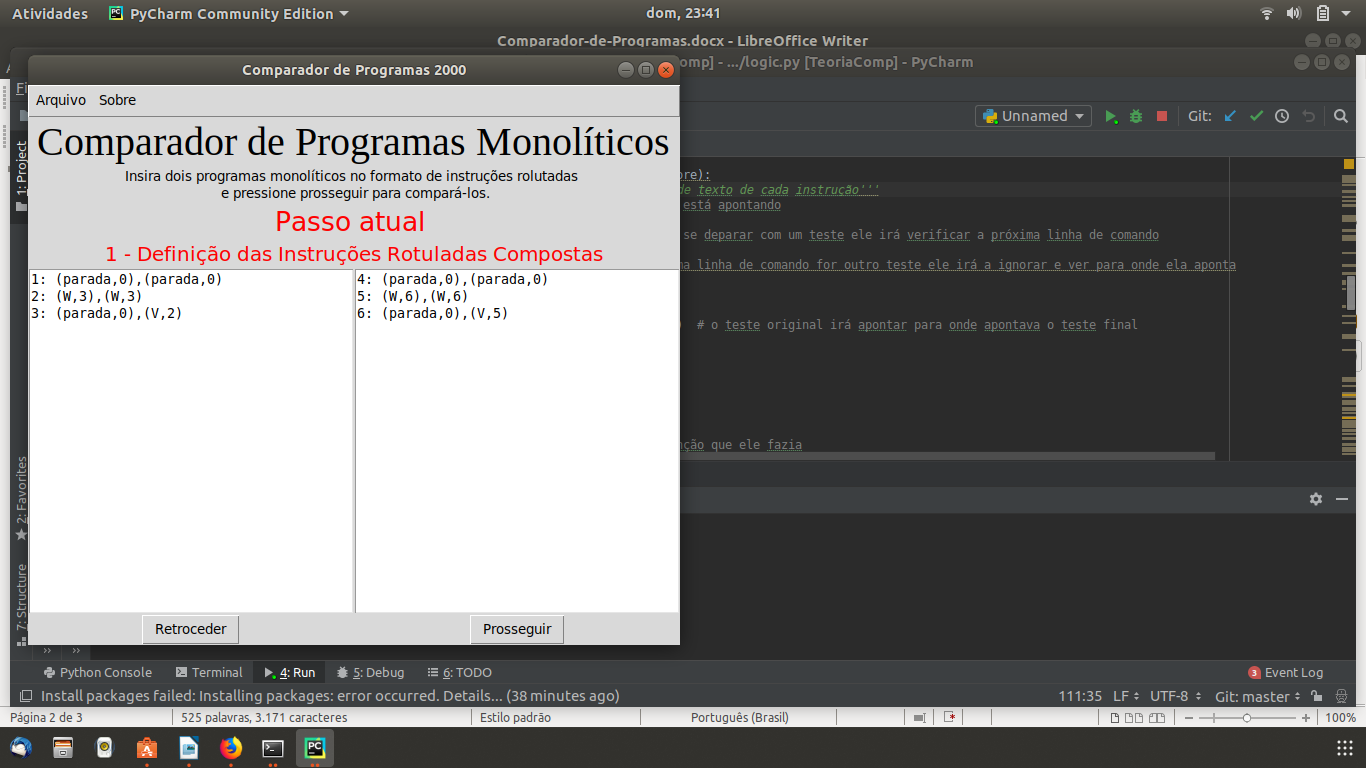
O aplicativo é capaz de traduzir programas monolíticos simples em programas monolíticos compostos. Após a etapa de tradução, o resultado é exibido na tela para que o usuário possa visualizar a conversão. Os programas, são então comparados exibindo o passo a passo de sua comparação até que o veredito seja dado pelo programa com a exibição dos resultados da comparação.

**2. Modo de usar**

Há dois campos em brancos na primeira tela exibição. No campo da esquerda você pode digitar o primeiro programa monolítico simples e, no campo da esquerda, você pode digitar o segundo programa monolítico simples, ambos precisam estar digitados para que o programa possa continuar. Quando os dois estiverem digitados você poderá pressionar o botão de Prosseguir. Ainda, se for de sua preferência, você pode carregar os programas por meio do Arquivo>Abrir Programas.

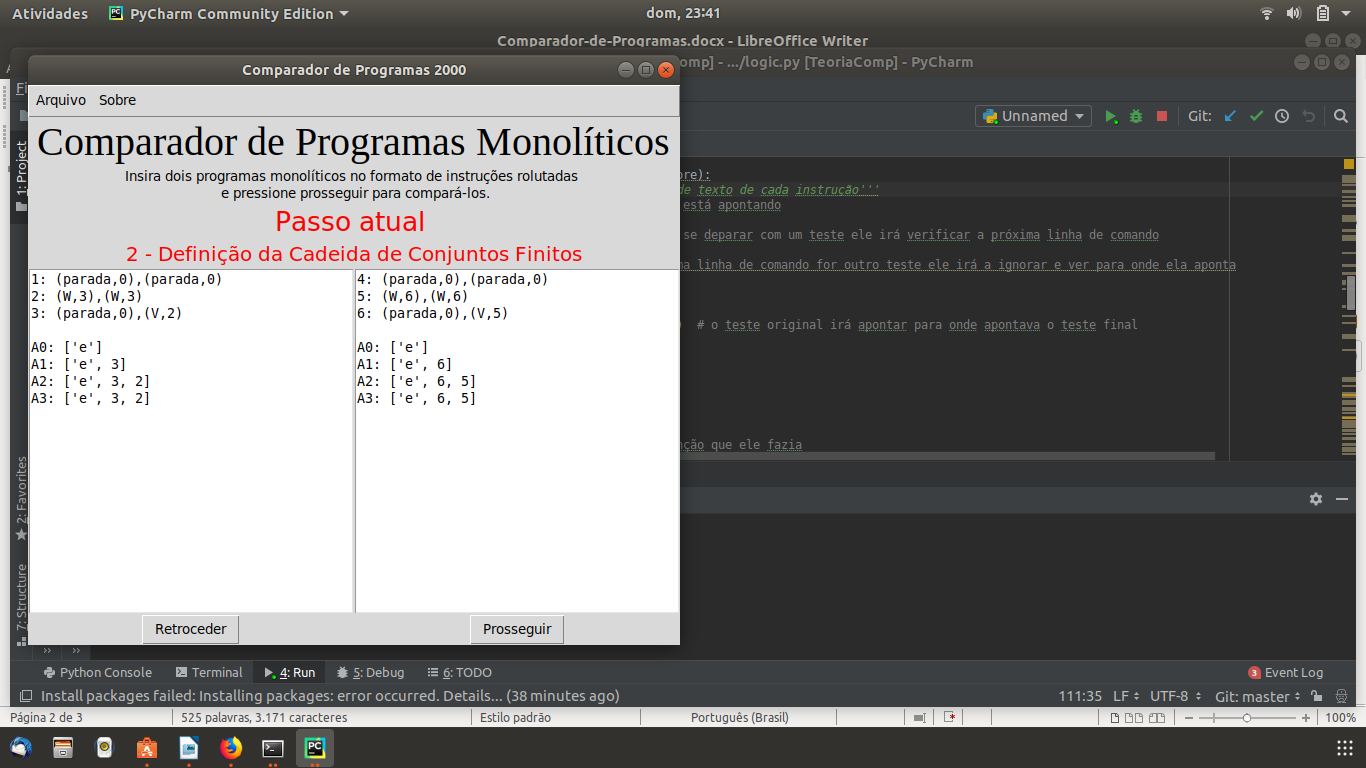
Uma tela irá se abrir para que você possa selecionar onde estão os arquivos de texto, um para cada programa. Em seguida, você poderá acompanhar todos os passos da comparação, tendo total liberdade de avançar e retroceder para o passo desejado, até mesmo podendo voltar ao início do programa para alterar algum dos programas. Para continuar você precisa pressionar o botão “Prosseguir”.

*Imagem 1. Estágio inicial do programa.*

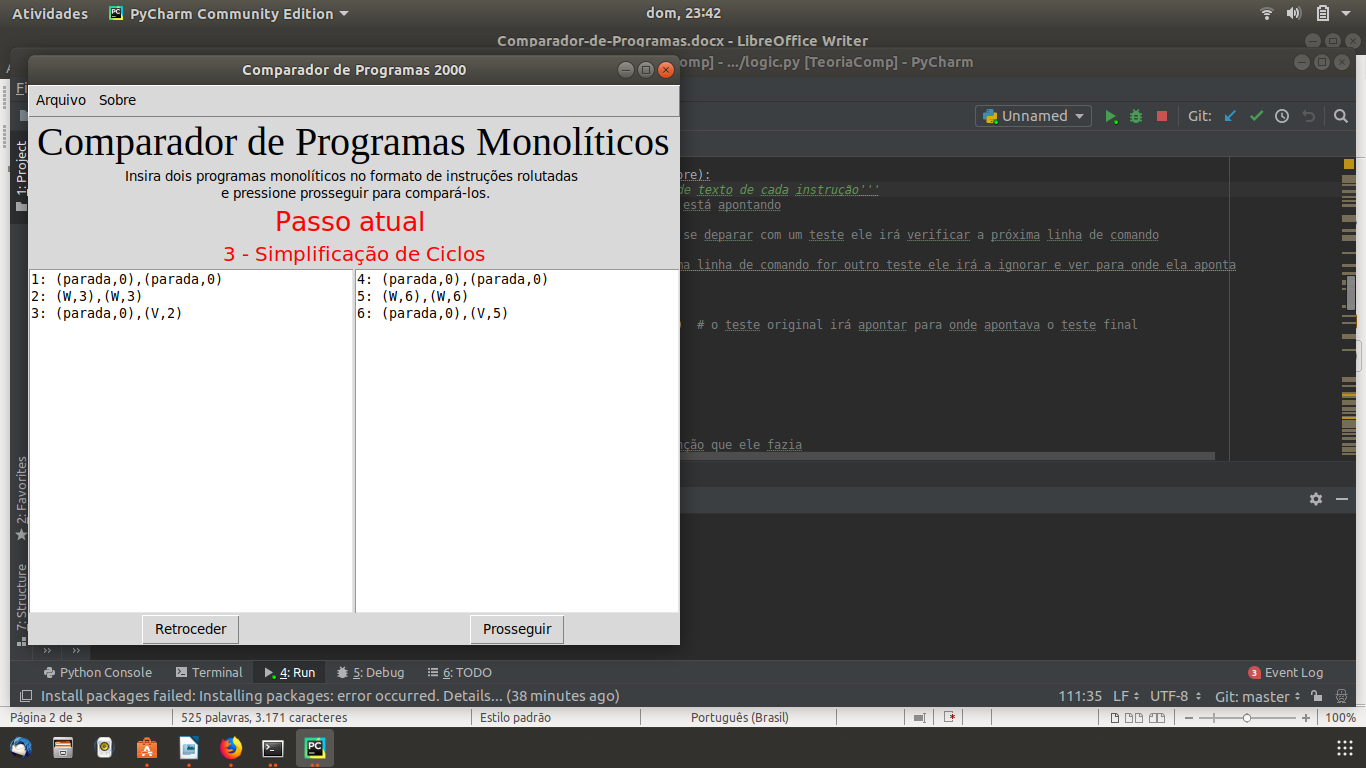
Na segunda tela, os programas são apresentados em sua forma composta. Caso queira continuar o passo a passo você pode pressionar o botão de “Prosseguir” e caso queira voltar ao passo anterior você pode pressionar o botão de “Retroceder”.

*Imagem 2. Primeiro estagio de comparaçao*

Na terceira tela, os programas são começam a ser simplificados. Caso queira continuar o passo a passo você pode pressionar o botão de “Prosseguir” e caso queira voltar ao passo anterior você pode pressionar o botão de “Retroceder”.

*Imagem 3. Segundo estagio de comparaçao.*

Na quarta tela, os programas são apresentados em sua forma simplificada. Caso queira continuar o passo a passo você pode pressionar o botão de “Prosseguir” e caso queira voltar ao passo anterior você pode pressionar o botão de “Retroceder”.

*Imagem 4. Terceiro estagio de comparaçao.*

**3. Padrão de entrada de dados**

Em relação as instruções do programa há um certo padrão que precisa ser seguido. Sempre comece a linha com uma instrução “Se” ou “Faça” dê espaço uma e apenas uma vez, logo em seguida defina uma palavra para nomear instrução e encerre com espaço. Depois escreva “para” e dê espaço uma e somente uma vez seguido de um número, sinalizando a linha do programa monolítico simples em que a instrução apontará. Se for um teste (ou seja, “se”), você precisa dar espaço e escrever novamente um “para” seguido de um e somente um espaço. Após isso, você pode sinalizar o número para qual a instrução apontará em caso de o teste retornar falso.

Ainda sobre o padrão de entrada dos dados é importante salientar que:

* Um teste não pode apontar para ele mesmo, caso contrário, o aplicativo acusará erro.
* Uma instrução de teste (ou seja, do tipo “se”) não pode apontar para o mesmo lugar em ambos os casos de verdadeiro de falso, caso contrário o aplicativo acusará erro.
* Nenhum dos campos de digitação de programas pode estar vazio, caso contrário, o aplicativo acusará erro.
* Uma instrução de execução de função (ou seja, do tipo “para”) não pode apontar para ela mesma, é necessário que uma segunda função que aponte para a primeira.