

Prova Dois

Manual de usuário do Interpretador

Santiago Quintero Hincapié

11726111

PCS 3216

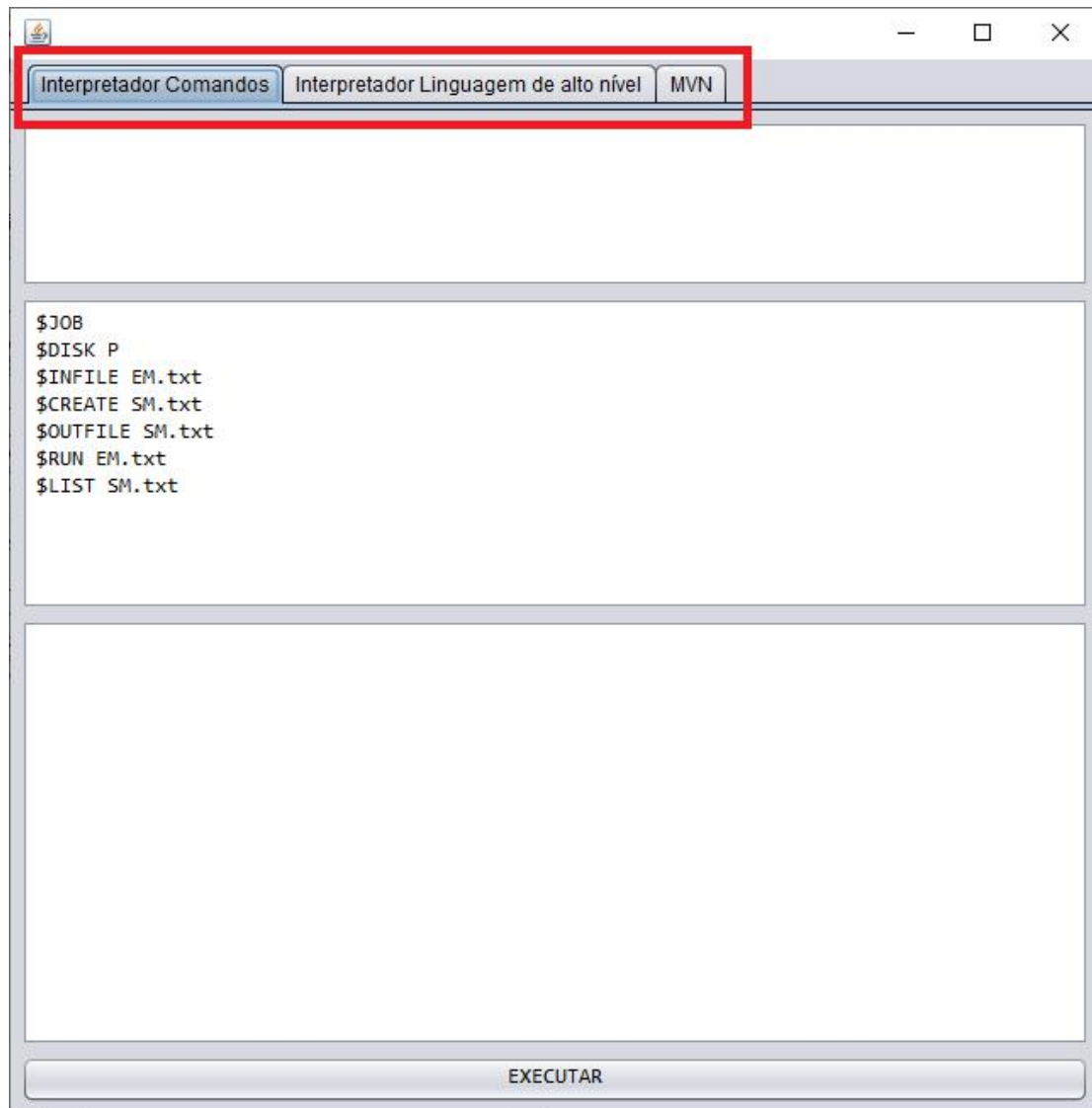
USP - POLI

Índice

Descrição da tela principal.....	2
Interpretador de Comandos.....	3
Tabela de Comandos.....	3
Tela do interpretador de comandos.....	4
1. Comandos lidos.....	4
2. Panel de escritura do usuário.....	4
3. Resposta do sistema.....	4
4. Executar.....	5
Interpretador da linguagem de alto nível.....	6
Formato dos comandos.....	6
Tela do interpretador da linguagem de alto nível.....	7
5. Panel de escritura do usuário.....	7
6. Resposta do sistema.....	7
7. Entrada.....	8
8. Executar.....	8
9. Clean Código.....	8
10. Clean all.....	8
MVN.....	9

Descrição da tela principal

Quando começa a aplicação se mostra um código já feito, e explicado no documento Teste do Interpretador, que pode ser utilizado para testar a aplicação ou pode ser apagado para escrever outro código, deste jeito a tela fica assim:



Na parte superior pode-se ver três telas diferentes, especificadas adiante.

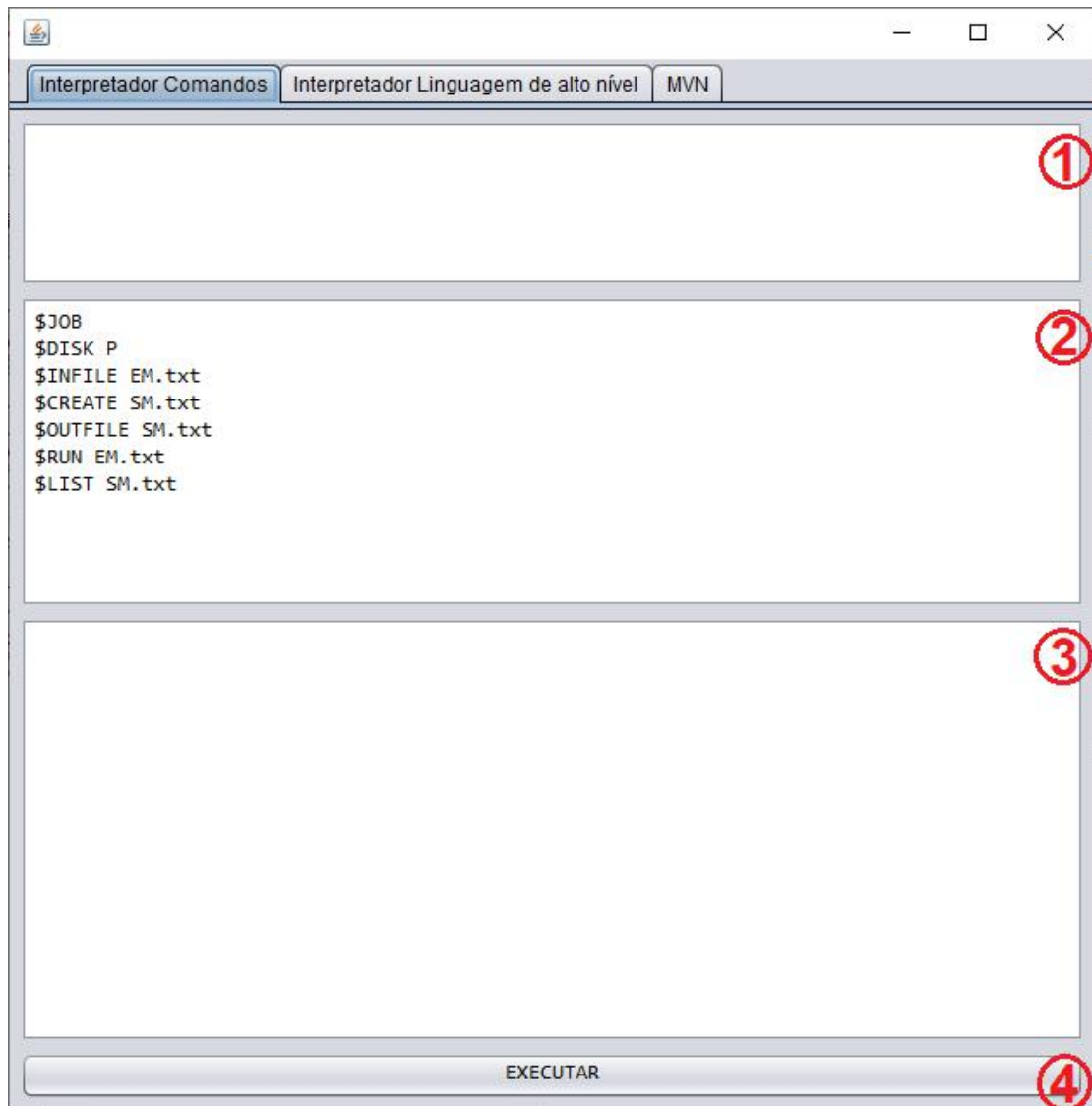
Interpretador de Comandos

Este interpretador esta encarregado de ler e executar os comandos de controle feitos pelo usuário.

Tabela de Comandos

comando	formato	ação
Inicia operação	\$JOB nome	Login - inicia um novo job, reinicia todo o sistema
Escolhe pasta	\$DISK pasta	Simula disco do sistema nesta pasta do hospedeiro
Arquivos existentes	\$DIRECTORY	Lista o conteúdo da pasta do hospedeiro
Cria arquivo	\$CREATE nome	Cria no disco um novo arquivo, se ainda não existir
Apaga arquivo	\$DELETE nome	Remove do disco o arquivo indicado, se existir
Mostra conteúdo	\$LIST nome	Apresenta o conteúdo do arquivo indicado, se existir
Mídia de entrada	\$INFILE nome	Adota o arquivo indicado como a fita de entrada
Mídia de saída	\$OUTFILE nome	Adota o arquivo indicado como a fita de saída
Arquivo em disco	\$DISKFILE nome	Adota o arquivo indicado como arquivo em disco
Executa programa	\$RUN nome	Executa o programa indicado (de sistema ou usuário)
Encerra operação	\$ENDJOB nome	Finaliza pendências e termina o job corrente

Tela do interpretador de comandos



1. Comandos lidos

Nesta secção vão ser escritos os comandos que foram interpretados corretamente.

2. Panel de escritura do usuário

Nesta secção o usuario pode escrever os comandos que vão ser lidos e executados pelo interpretador, se o usuario tiver algum erro de escritura o interpretador vai detener a execução e especificar na secção 3 qual é o erro.

3. Resposta do sistema

Nesta secção o sistema vai especificar a correta leitura dos comandos ou se algum comando tem erro de escritura, algumas resposta do sistema podem ser:

- novo job, sistema reiniciado
- Agora o disco do sistema é: 'pasta'
- Conteúdo do arquivo 'EM.txt' foi copiado no interpretador de linguagem de alto nível, para ser trabalhado
- Arquivo 'SM.txt' criado
- Saída do interpretador da linguagem de alto nível será salvo no arquivo 'SM.txt'
- O interpretador da linguagem de alto nível executó o arquivo 'EM.txt'
- E assim por diante

4. Executar

Este botão executa os comandos escritos pelo usuario no panel se escritura.

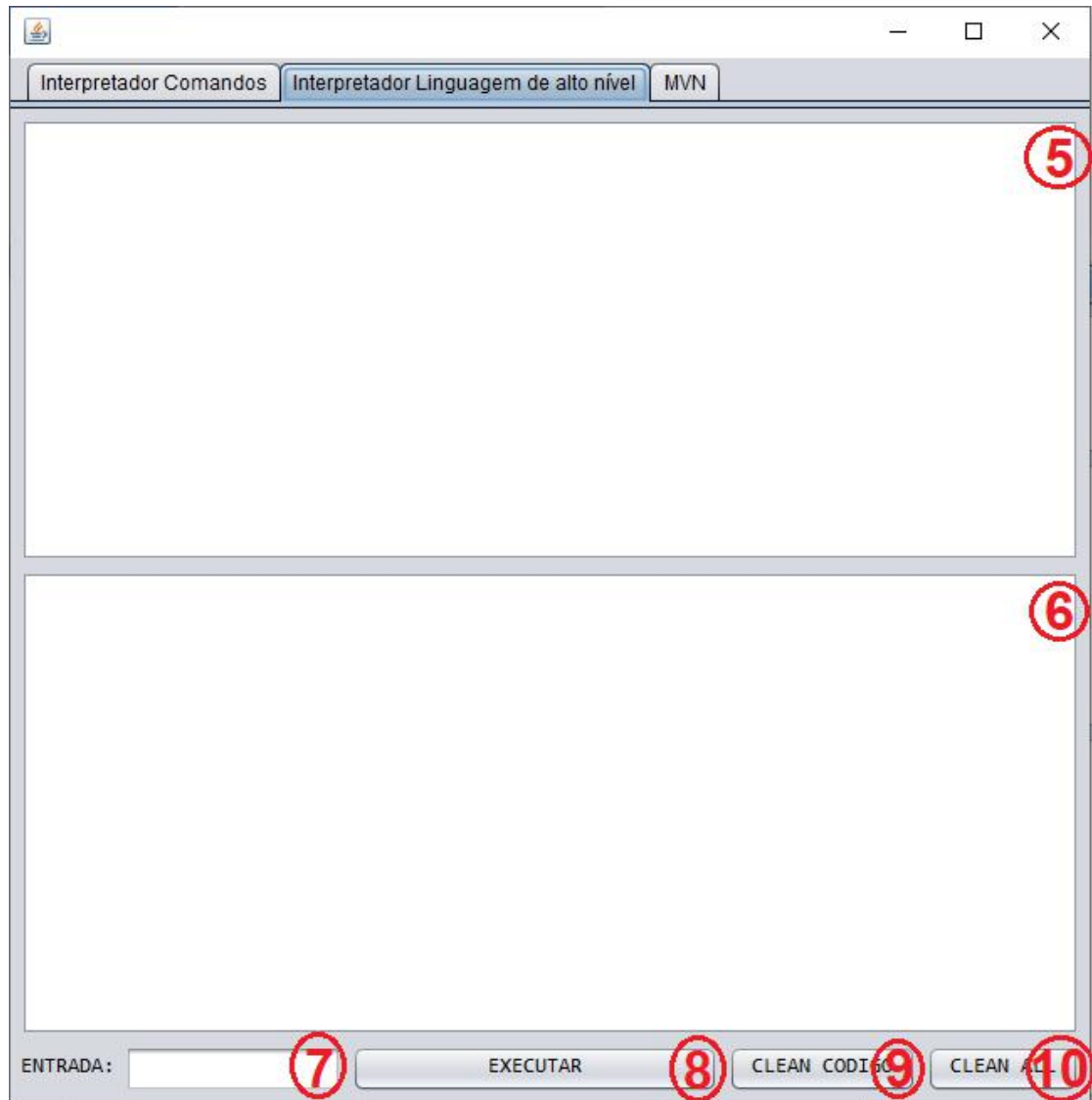
Interpretador da linguagem de alto nível

Este interpretador esta encarregado de ler e executar a linguagem de alto nível feita pelo usuário.

Formato dos comandos

Comando	Formato sintático	Operação
Comando com rótulo	NNNN : comando	Associa rótulo(s) NNNN ao comando
Comando sem rótulo	comando	Este comando não tem rótulo
Comando atribuição	LET NNNN = expressão	Associa a NNNN o valor da expressão
Comando desvio incondicional	GOTO NNNN	Desvia para o rótulo NNNN
Comando desvio condicional	GOTO NNNN IF expressão op expressão	Desvia para o rótulo NNNN apenas se essa comparação for verdadeira
Comando entrada	READ NNNN	Dá a NNNN o valor lido na entrada
Comando saída	WRITE expressão	Grava na saída o valor da expressão
Expressão	valor {(+ - * /)} valor	Série de operações sobre valores
valor	NNNN DDDD	Valores são: ou variáveis ou números

Tela do interpretador da linguagem de alto nível



5. Panel de escritura do usuário

Nesta secção o usuario pode escrever o código que vão ser lidos e executados pelo interpretador, se o usuario tiver algum erro de escritura o interpretador vai detener a execução e especificar na *secção 6* qual é o erro.

6. Resposta do sistema

Nesta secção o sistema vai especificar a correta leitura do código ou se tiver algum erro de escritura, algumas resposta do sistema podem ser:

- Variável A=1 salva
- Variável MAIOR=3 atualizada
- Desvio desde 10 a 11 feito

- Código compilado correctamente
- E assim por diante

7. Entrada

Esta caixa de texto vai ser utilizada pelo usuario se no código escrever-se '*READ NNNN*'. A execução do código vai detener-se e esperar que o usuario escreva o nome de uma variável o um numero entero para ser asignado a '*NNNN*'.

8. Executar

Este botão executa o código escritos pelo usuario no panel de escritura.

9. Clean Código

Este botão limpa o panel de escritura, variáveis e rótulos salvos para que o usuario escreva e execute um código novo.

10. Clean all

Este botão limpa tudo, exceto o '*\$OUTFILE*' especificado no interpretador de comandos.

MVN

Esta parte da prova foi desenvolvida e explicada anteriormente na materia, por esse motivo so vou explicar que foi utilizado, se quiser ver o manual do usuario deste projeto só, dar [click aqui](#).

A parte marcada em vermelho representa a memoria do sistema e é utilizada pelo interpretador do comandos com '\$DISKFILE NNNN' onde 'NNNN' é o nome de um arquivo .txt que contem a memoria para ser carregada no MVN.

