

Prova Dois

Teste do Interpretador

Santiago Quintero Hincapié

11726111

PCS 3216

USP - POLI

Índice

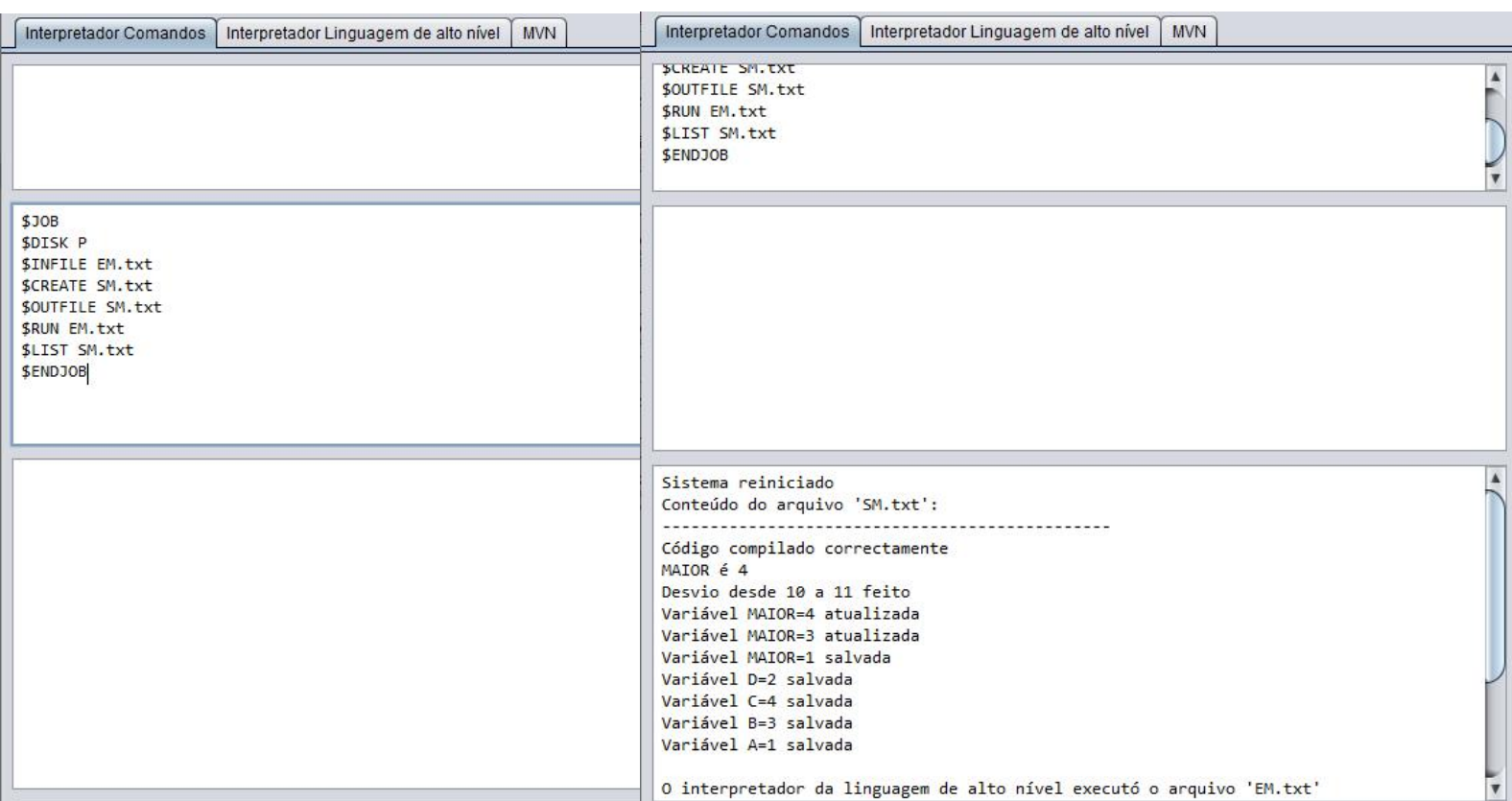
Teste da integração do sistema.....	2
Teste da Linguagem de comandos.....	2
Comando: \$JOB.....	3
Comando: \$DISK P.....	4
Comando: \$INFILE EM.txt.....	5
Comando: \$CREATE SM.txt.....	7
Comando: \$OUTFILE SM.txt.....	8
Comando: \$RUN EM.txt.....	9
Comando: \$LIST SM.txt.....	11
Comando: \$ENDJOB.....	12
Teste da Linguagem de programação.....	13

Teste da integração do sistema

O Código de comandos que vai ser utilizado para o teste, já esta carregado no executable para ser tentado ou apagado se o usuário quiser, o disco 'P' também esta criado apara este teste e dentro dele esta o arquivo 'EM.txt' que contem código em linguagem de alto nível para testar o interpretador da linguagem de alto nível.

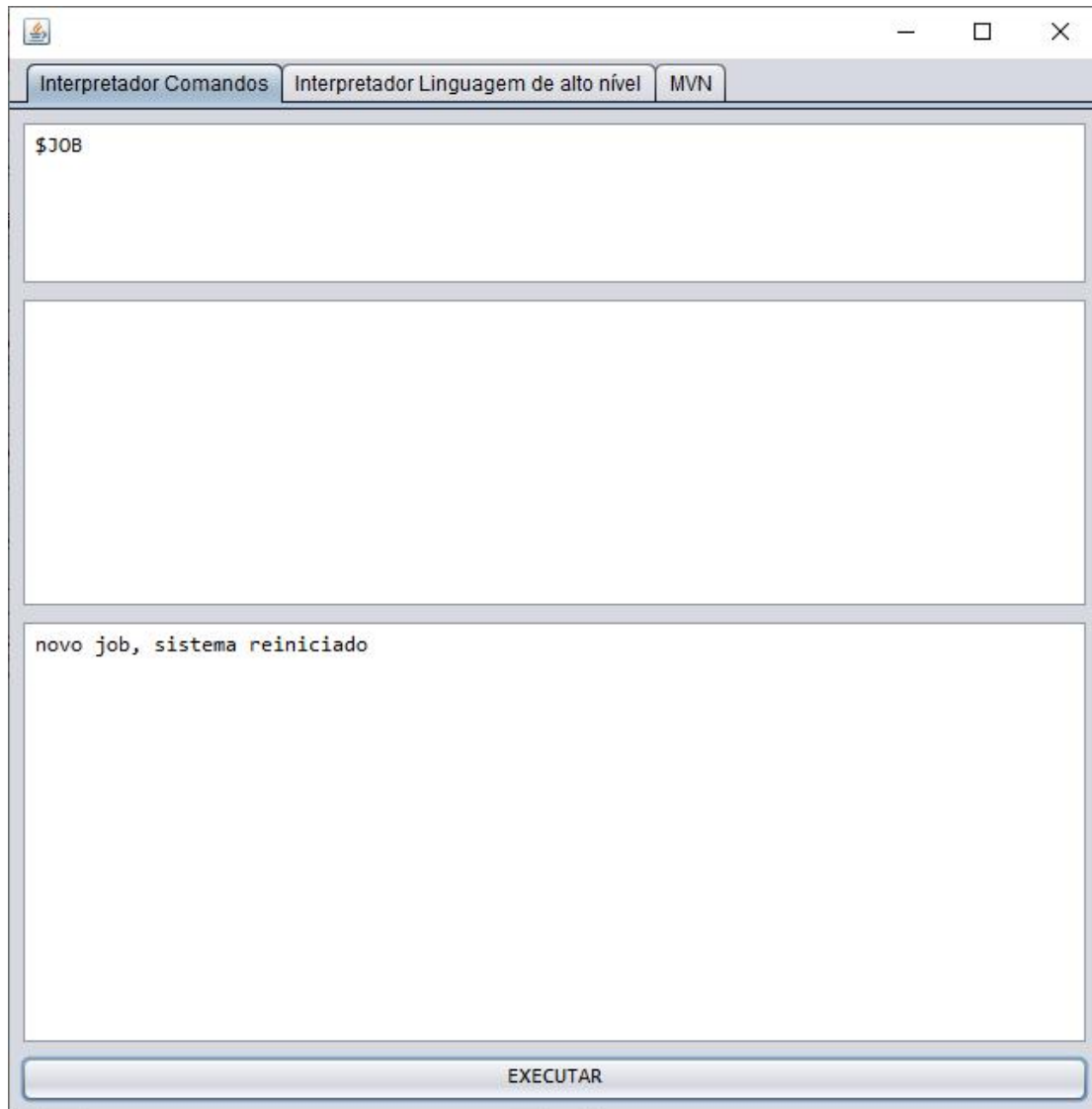
Teste da Linguagem de comandos

Nas dois Imagens abaixo pode se ver os comandos que vão ser executados (especificado adiante) e a resposta do sistema.



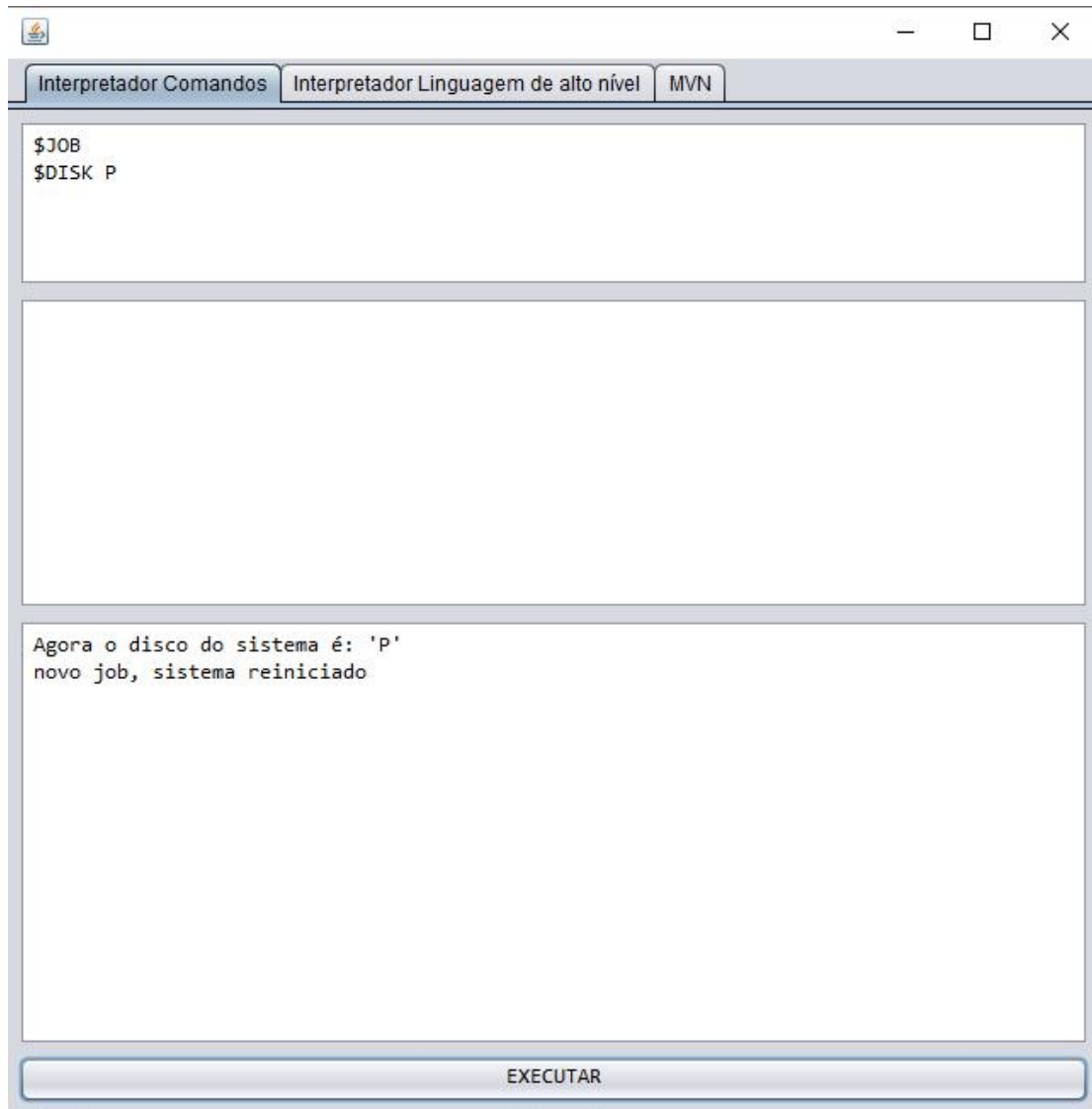
Comando: \$JOB

Preparação para início de funcionamento do sistema, falta especificar o disco no qual vai-se trabalhar.



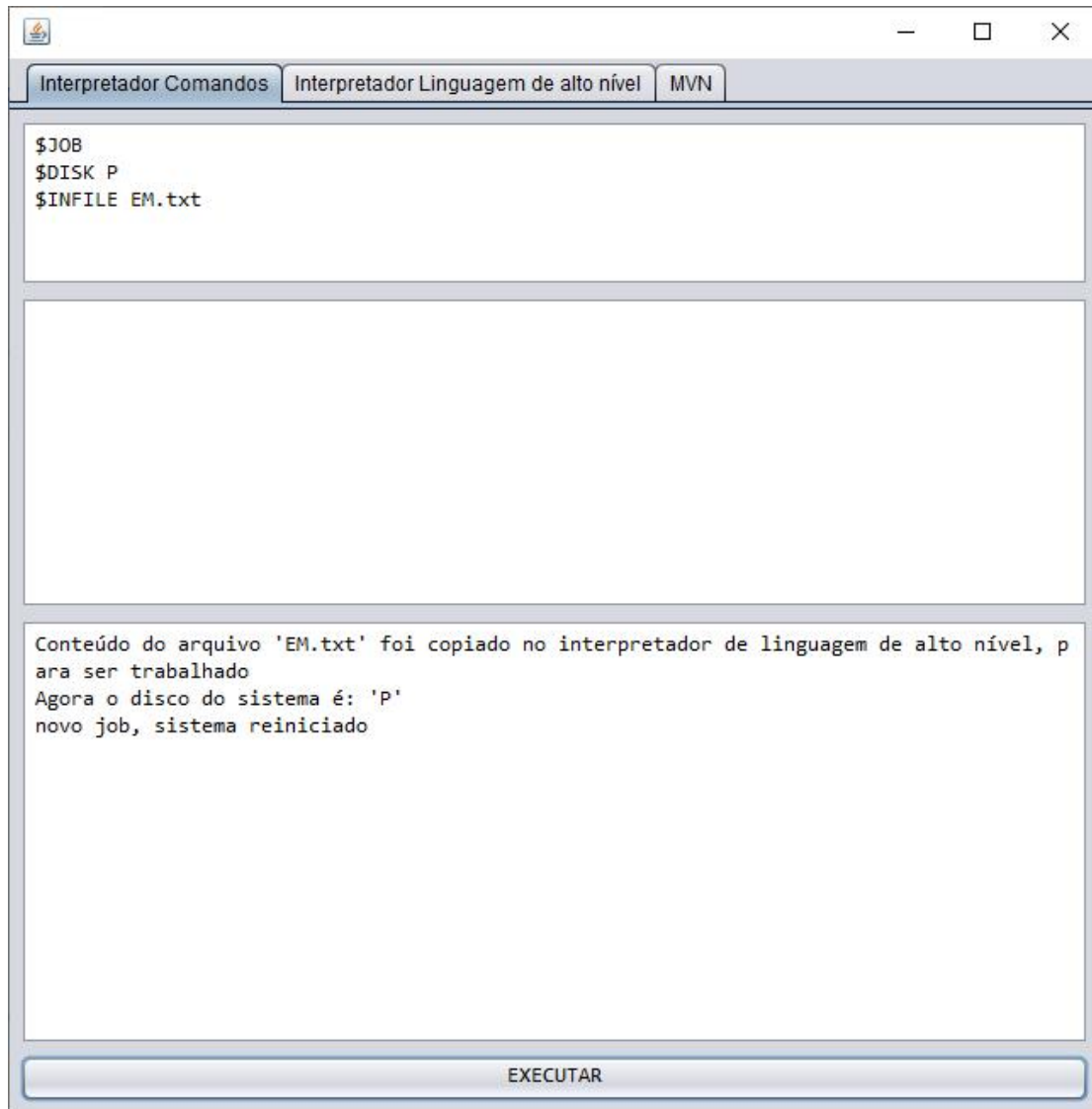
Comando: \$DISK P

Especificação do disco com o qual vai trabalhar o sistema, neste caso 'P' o qual foi criado para este teste e contem um arquivo .txt com linguagem de alto nível.

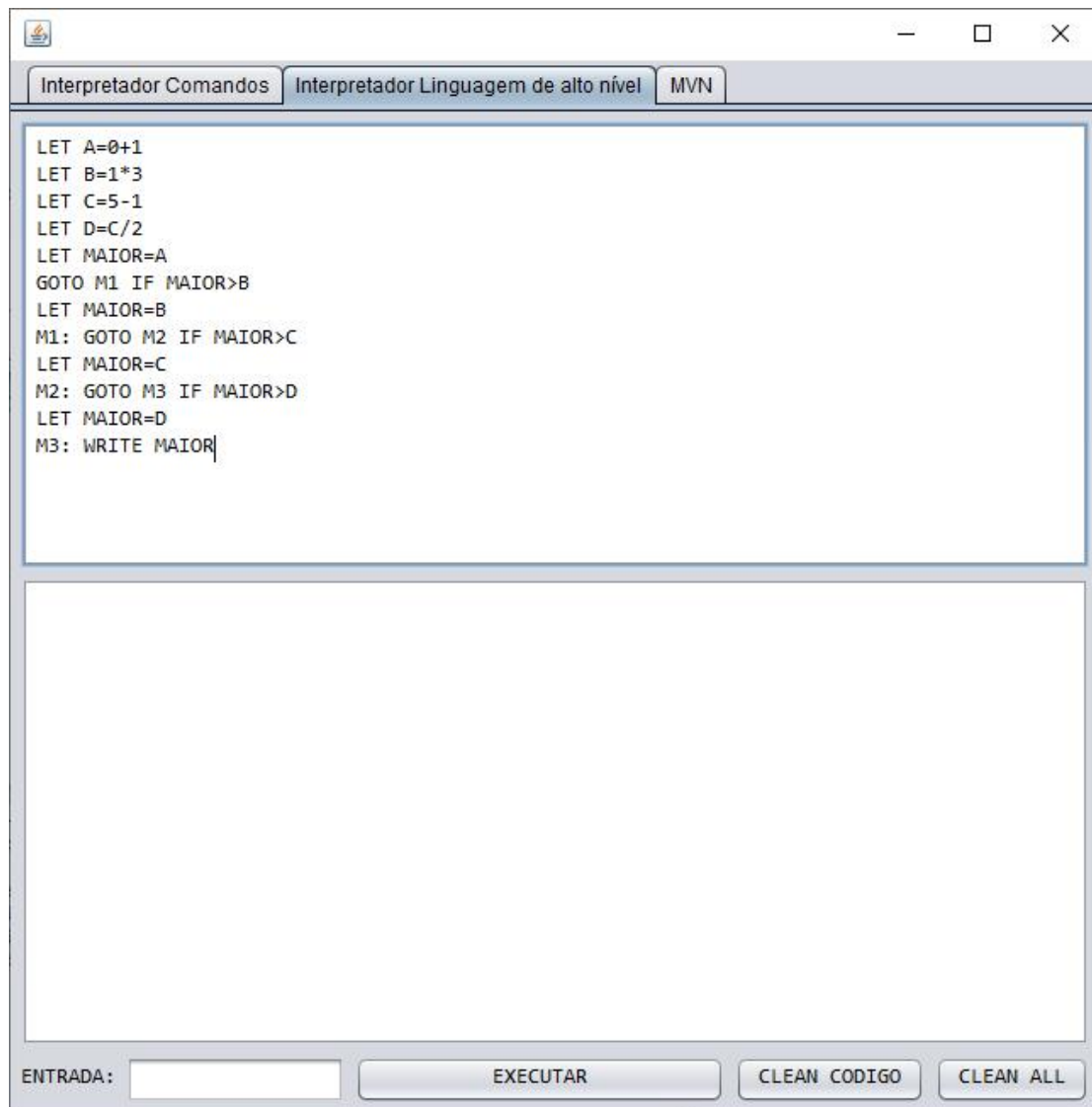


Comando: \$INFILE EM.txt

O arquivo EM.txt contém linguagem de alto nível, o qual foi carregado no interpretador de linguagem de alto nível para ser trabalhado.

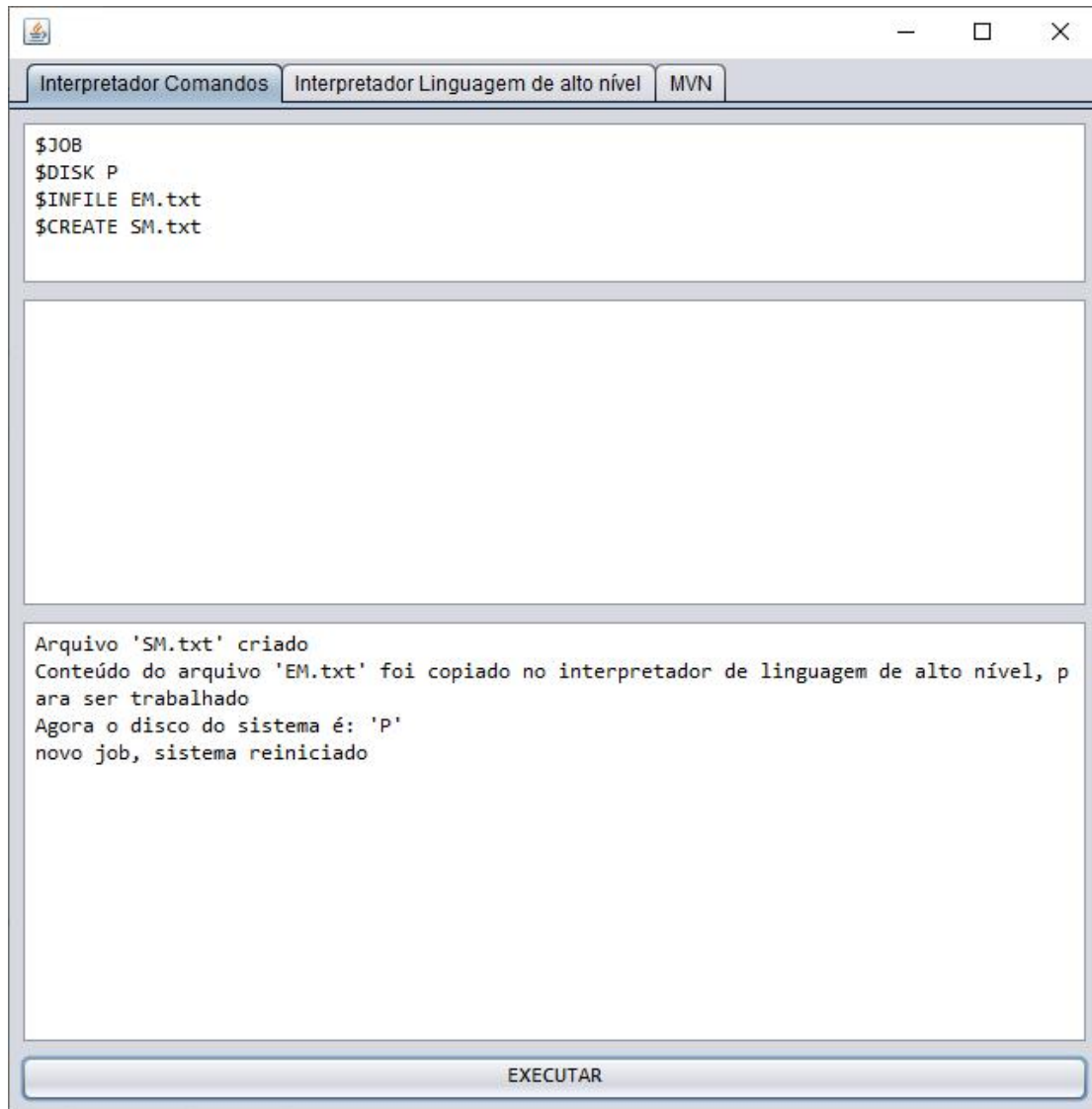


Aqui pode se ver a linguagem de alto nível carregada no interpretador.



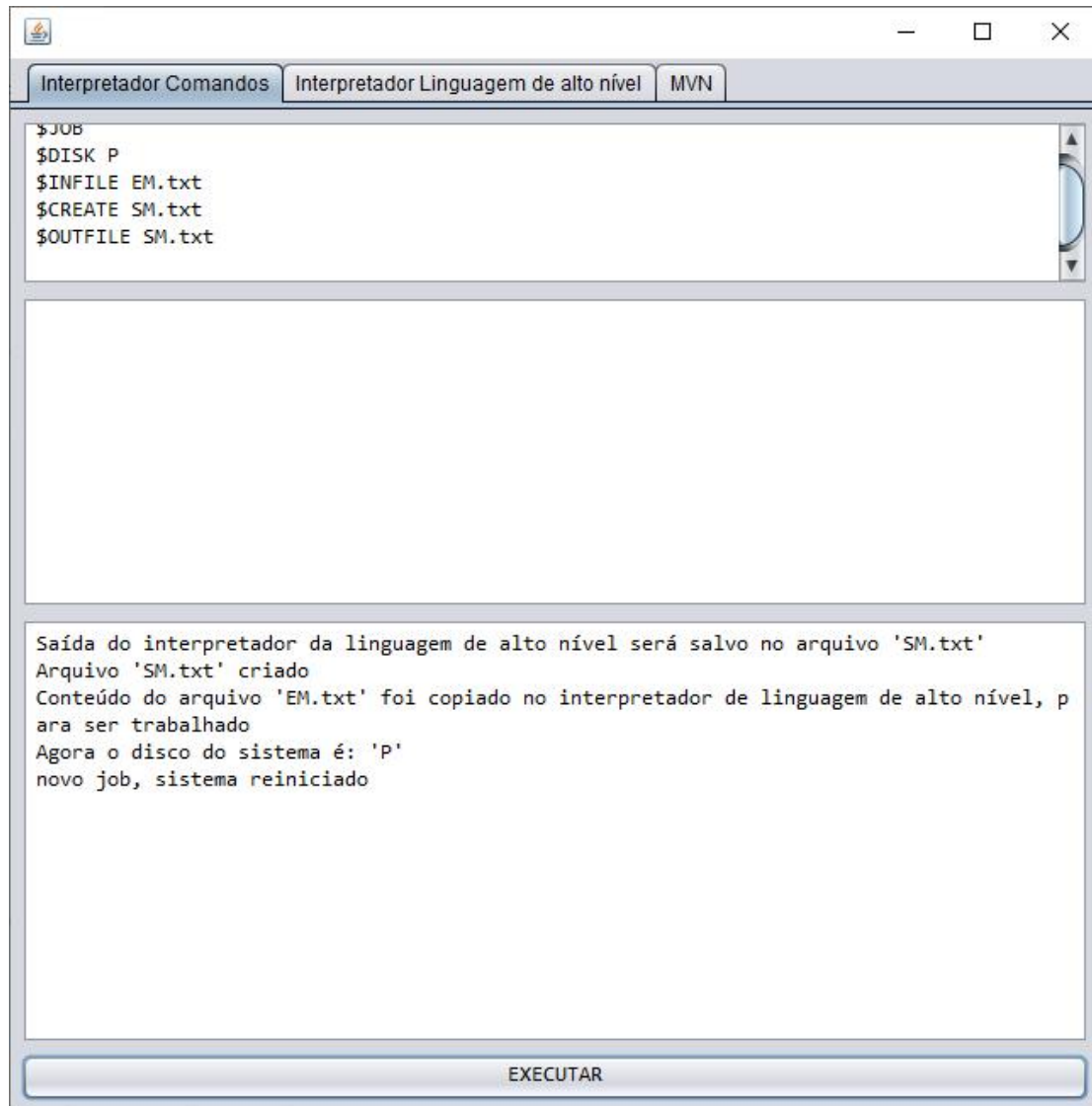
Comando: \$CREATE SM.txt

Com este comando criamos um arquivo .txt se ainda nao existir, para escrever as saídas da execução da linguagem de alto nível.



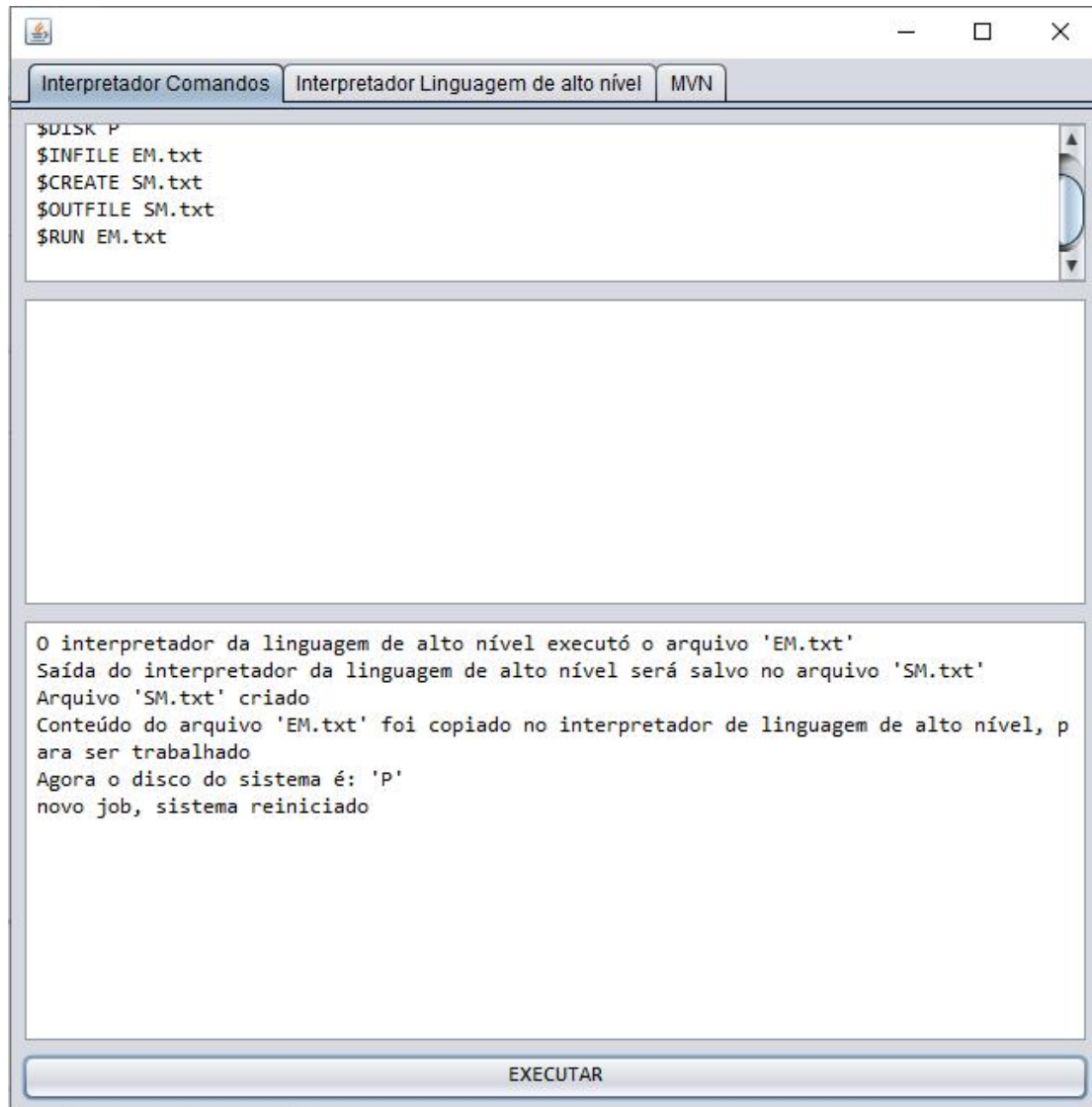
Comando: \$OUTFILE SM.txt

Especificação do arquivo onde vai-se salvar as saídas do interpretador da linguagem de alto nível.

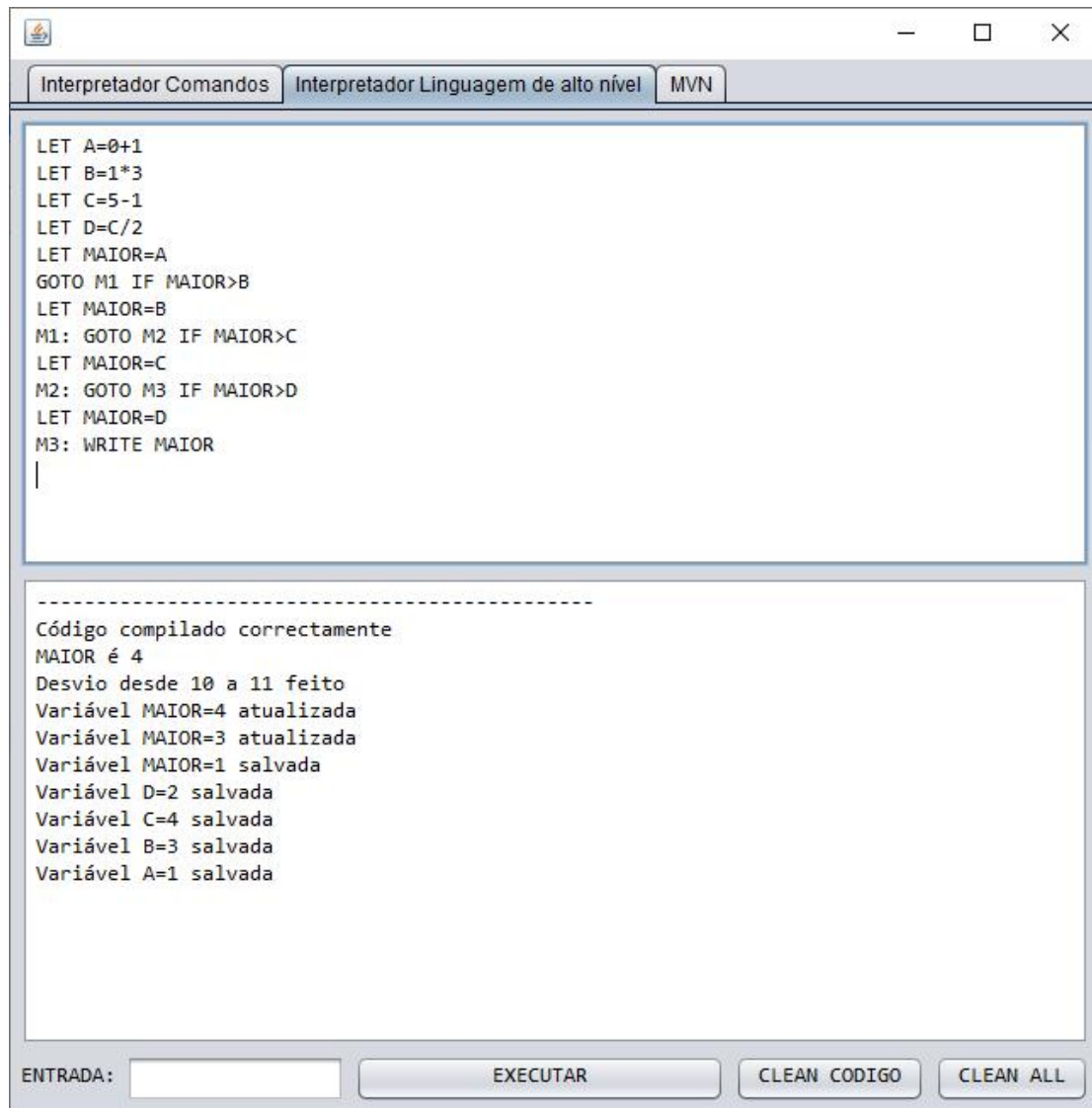


Comando: \$RUN EM.txt

Com este comando pode-se executar um arquivo no interpretador da linguagem de alto nível desde o interpretador de comandos.



Aqui pode-se ver a execução do código em linguagem de alto nível (especificado adiante).



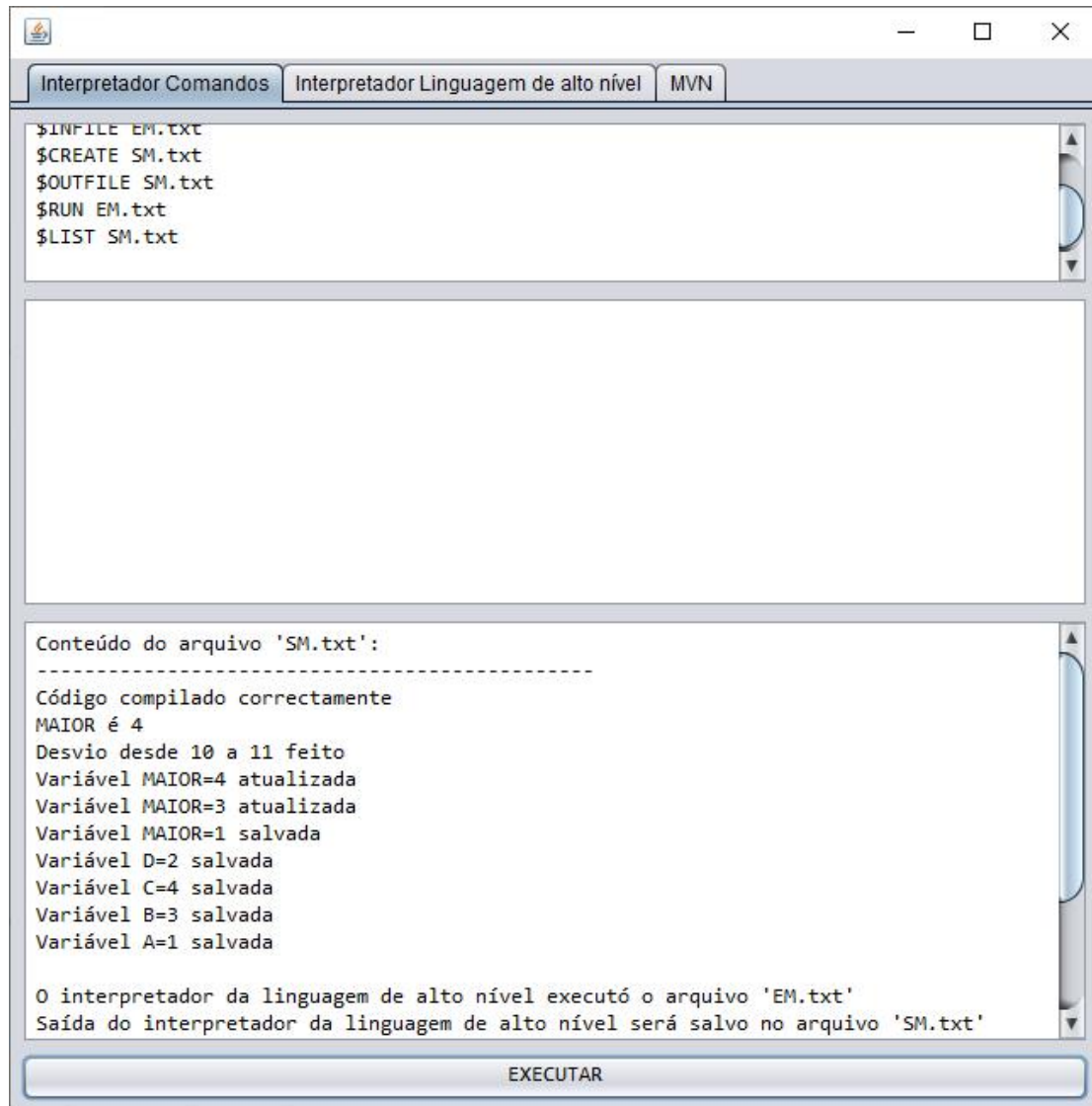
```
LET A=0+1
LET B=1*3
LET C=5-1
LET D=C/2
LET MAIOR=A
GOTO M1 IF MAIOR>B
LET MAIOR=B
M1: GOTO M2 IF MAIOR>C
LET MAIOR=C
M2: GOTO M3 IF MAIOR>D
LET MAIOR=D
M3: WRITE MAIOR
|
```

Código compilado correctamente
MAIOR é 4
Desvio desde 10 a 11 feito
Variável MAIOR=4 atualizada
Variável MAIOR=3 atualizada
Variável MAIOR=1 salva da
Variável D=2 salva da
Variável C=4 salva da
Variável B=3 salva da
Variável A=1 salva da

ENTRADA:

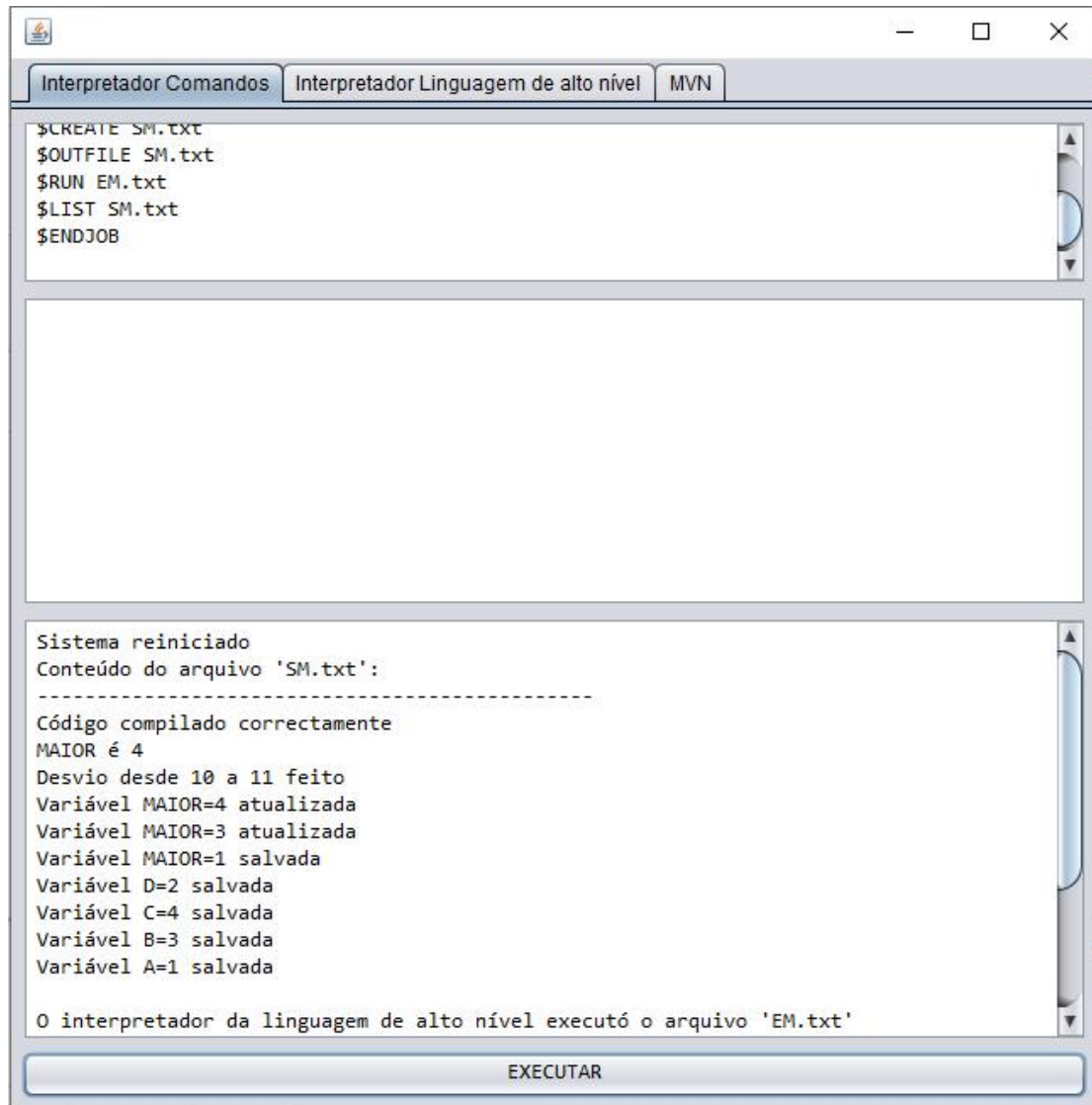
Comando: \$LIST SM.txt

Com este comando pode-se ver o conteúdo do arquivo no qual salvamos a saída do interpretador da linguagem de alto nível.



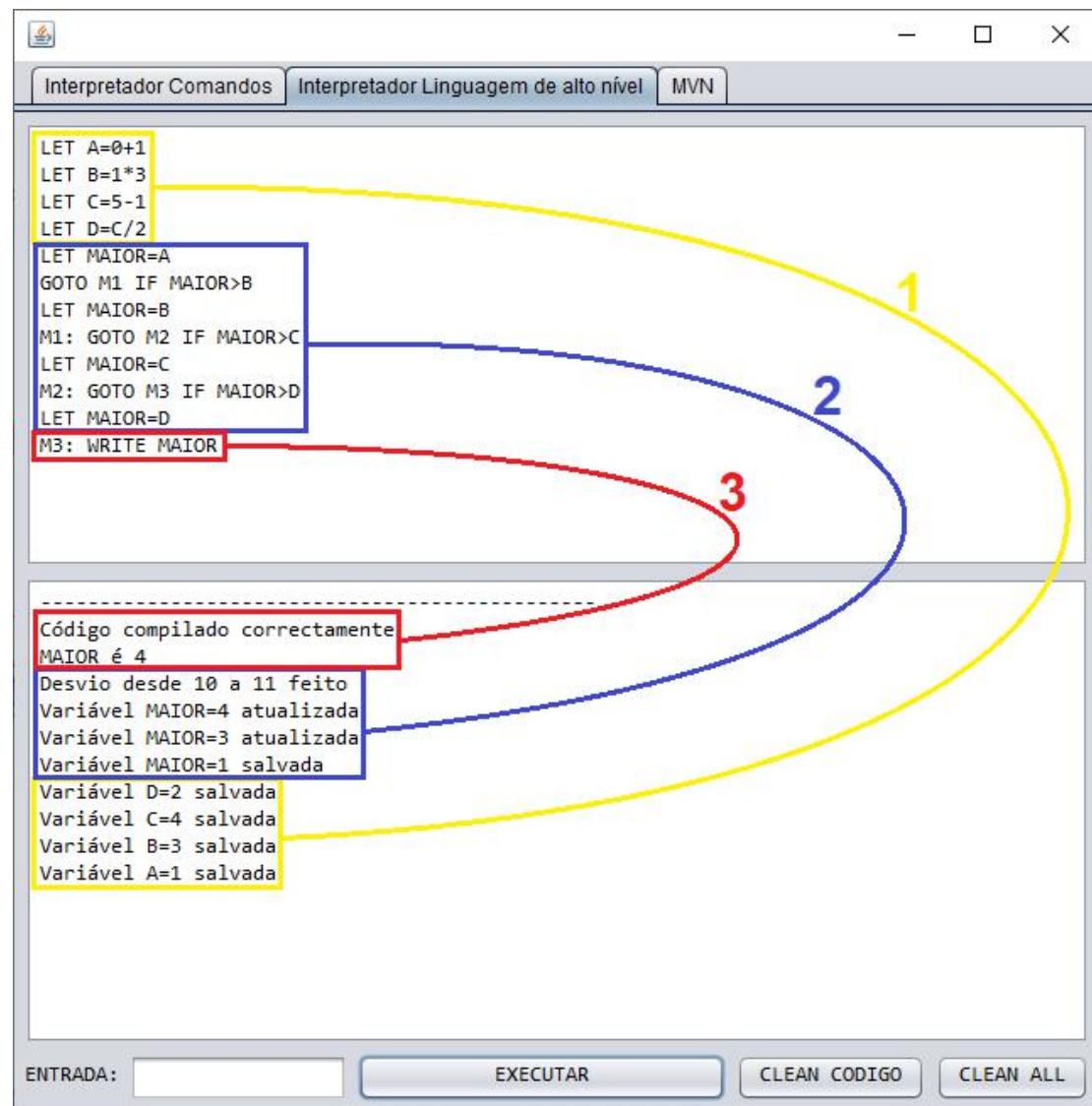
Comando: \$ENDJOB

Finalmente com este comando de reinicia o sistema para començar a trabalhar com uma nova sequência de comandos.



Teste da Linguagem de programação

Em geral este código identifica o numero maior de uma sequencia de números.



1. Na primeira parte de especifica as variáveis as quais vão conter os números que vão ser analisados, pode-se ver que se utiliza os comandos (LET), (+), (-), (*) E (/).
2. Nesta parte se procura qual es a variável que contem o numero maior e é salva na variável MAIOR, pode-se ver que se utiliza o comando (GOTO IF) e diferentes rótulos.
3. Finalmente de mostra qual é o numero maior.