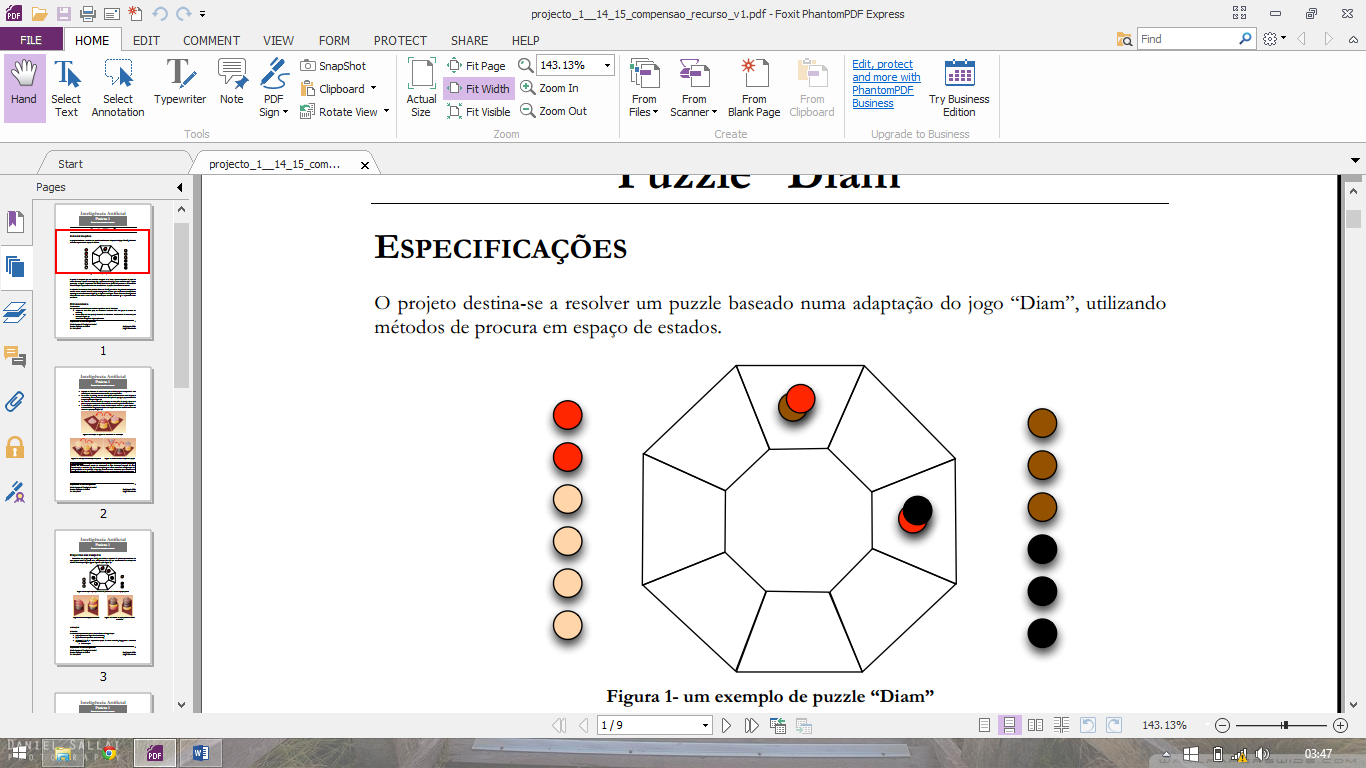
Manual de Utilizador

2º Projeto

“DIAM”



**Docentes**

Prof. Joaquim Filipe

Prof. Cedric Grueau

**Trabalho realizado por:**

Daniel Costa 120221058

Patricia Garcia 120221008

Conteúdos

[Objetivos do Programa 3](#_Toc428648736)

[Introdução ao “Diam” 3](#_Toc428648737)

[Descrição Geral de Funcionamento 3](#_Toc428648738)

[Instalação do Software LispWorks 4](#_Toc428648739)

[Download e Instalação 4](#_Toc428648740)

[Execução do Software LispWorks 5](#_Toc428648741)

[Limitações da Aplicação 10](#_Toc428648742)

# Objetivos do Programa

O objetivo de resolução deste puzzle é “fazer um *Diam*”, ou seja, colocar duas peças da mesma cor no mesmo nível de empilhamento, frente a frente.

Para “fazer um Diam”, o nível mais baixo - o nível 1 - não é tido em consideração; somente os níveis 2, 3 e 4 permitem fazer um Diam.

Para resolver este problema foram utilizados conhecimentos adquiridos na disciplina de Inteligência Artificial utilizando a linguagem de programação Lisp.

## Introdução ao “Diam”

O jogo “Diam” é um puzzle composto por um tabuleiro octogonal de 8 casas e quatro conjuntos de peças de cores diferentes: 4 peças de cor bege (B), 4 peças vermelhas (V), 4 peças castanhas (C) e 4 peças pretas (P).

Nesta versão do problema as peças é possível movimentar todas as peças no puzzle.

## Descrição Geral de Funcionamento

A aplicação é iniciada com um menu que fornece duas opções: resolução de um puzzle e regras do jogo.

A opção de resolução de um puzzle possibilita a escolha de um dos vários tabuleiros de jogo, apresentando em seguida três opções de algoritmos que permitem resolver os problemas propostos, dependendo do algoritmo escolhido, será apresentada a opção de escolha de profundidade no caso do algoritmo depth-first e a opção de seleção da heurística no caso do algoritmo A\*.

O funcionamento básico da aplicação consiste na escolha de um desses problemas, escolha de um algoritmo de resolução e observação de dados sobre a resolução e solução do puzzle como os nós gerados, os nós expandidos, o caminho percorrido até á solução, etc. todas estas informações são então armazenadas num ficheiro “estatísticas.dat”.

# Instalação do Software LispWorks

Os seguintes passos são para a instalar o software LispWorks, para sistema operativo Windows, necessário para executar este projeto.

## Download e Instalação

Para instalar a aplicação, deverá se registar em <http://www.lispworks.com/downloads/index.html>. Depois de escolher o sistema operativo adequado, poderá descarregar o ficheiro de instalação, carregando no botão “Proceed to Download” (Figura 1).

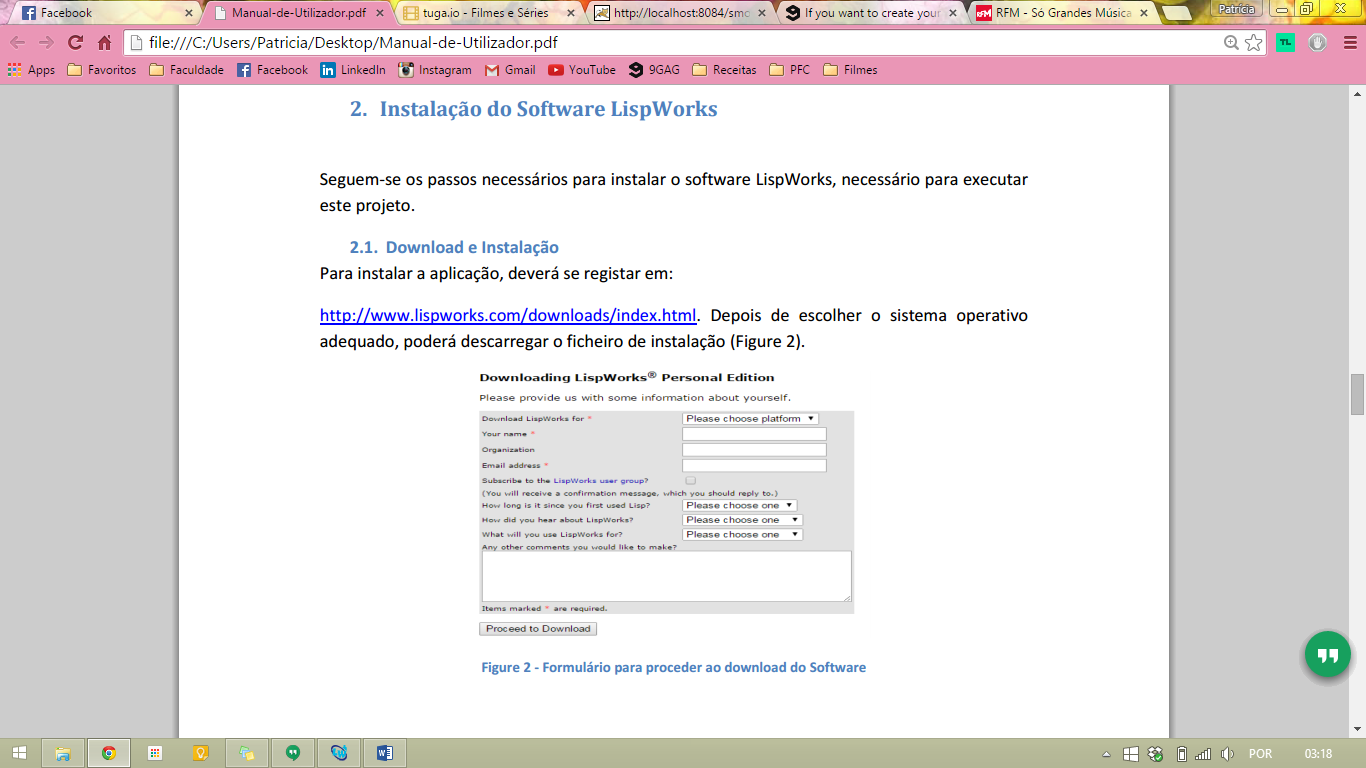


Figura 1- Lisp Instalação Software

Seguir os passos indicados na instalação do software LispWork.

# Execução do Software LispWorks

1. Para executar o programa, é necessário selecionar o ficheiro projecto.lisp através da opção: File->Open... (Figura 2.)

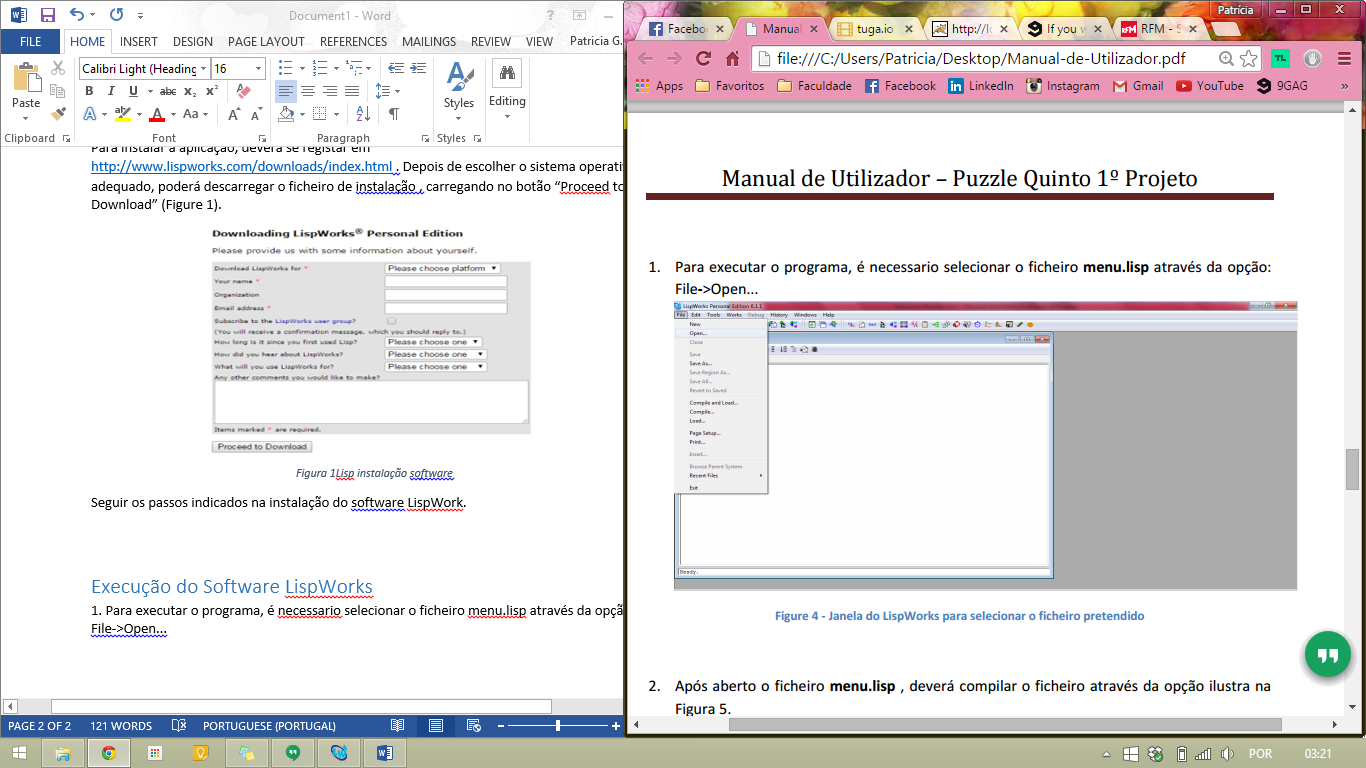


Figura 2 - Janela do LispWorks para selecionar o ficheiro pretendido

1. Após ter selecionado e aberto o ficheiro pretendido (que deverá encontrar-se numa pasta denominada Projeto situada na raiz do computador “C:\IA\P1…”), deverá compilar o ficheiro através da opção que mostra a seguinte figura (Figura 3.)

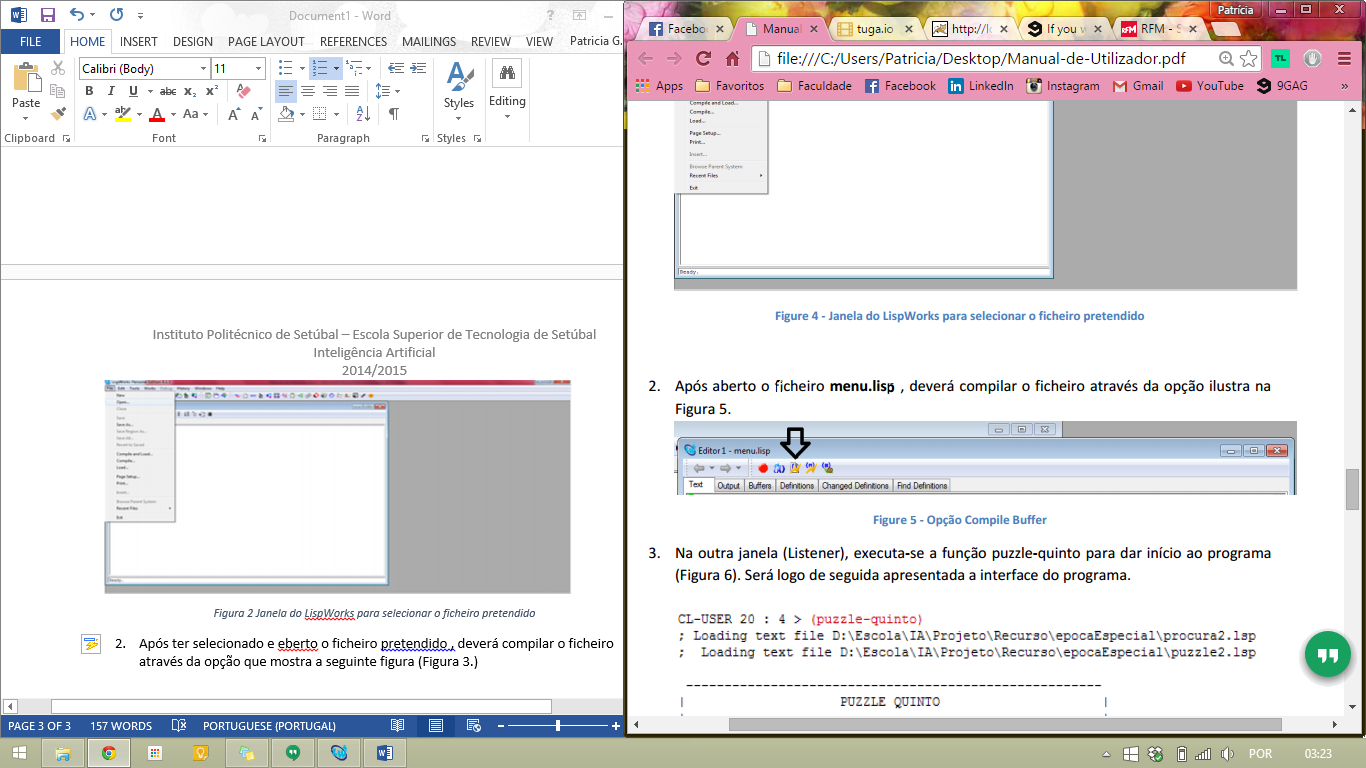


Figura 3 - Opção Compile Buffer

1. Após ter compilado o ficheiro, na janela (*Listener*), executa-se a função **(menu-principal)** para dar início ao programa, onde é apresentado o menu que permite resolver um puzzle ou ver as regras do jogo. (Figura 4)

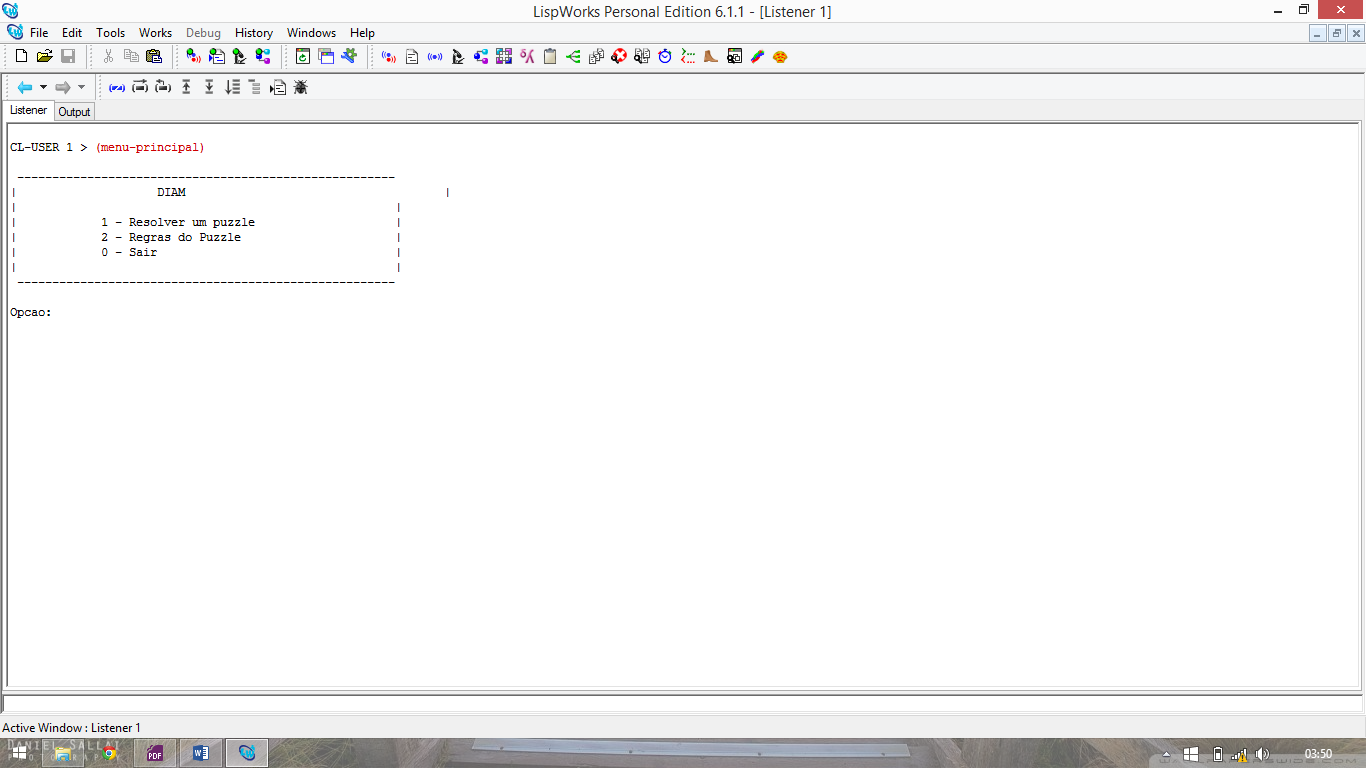


Figura 4 - Interface Inicial

1. Se se selecionar opção 2 serão apresentadas as regras do jogo DIAM. (Figura 5.)

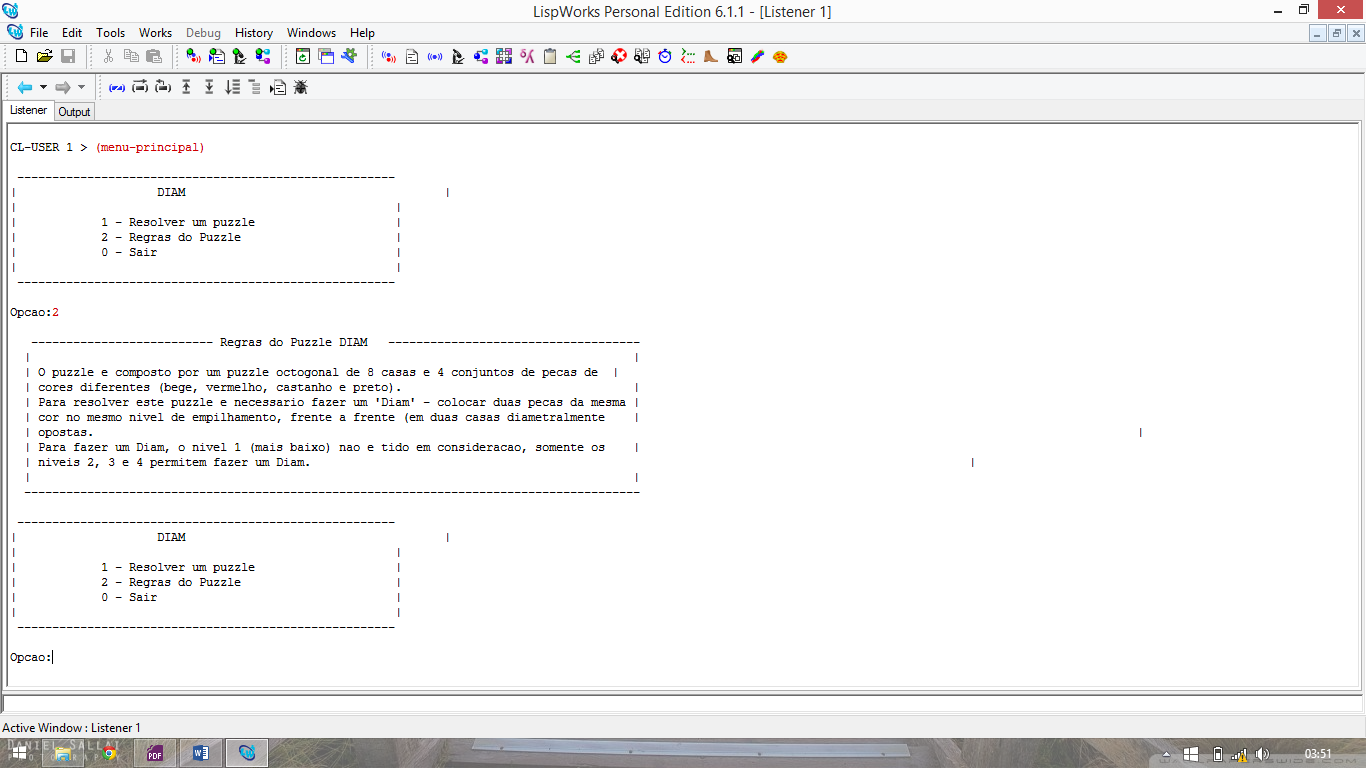


Figura 5 - Interface Regras Jogo

1. Surgirá novamente o menu e desta vez vamos selecionar a opção 1 para resolver um problema. Serão apresentados os puzzles disponíveis para o utilizador escolher.

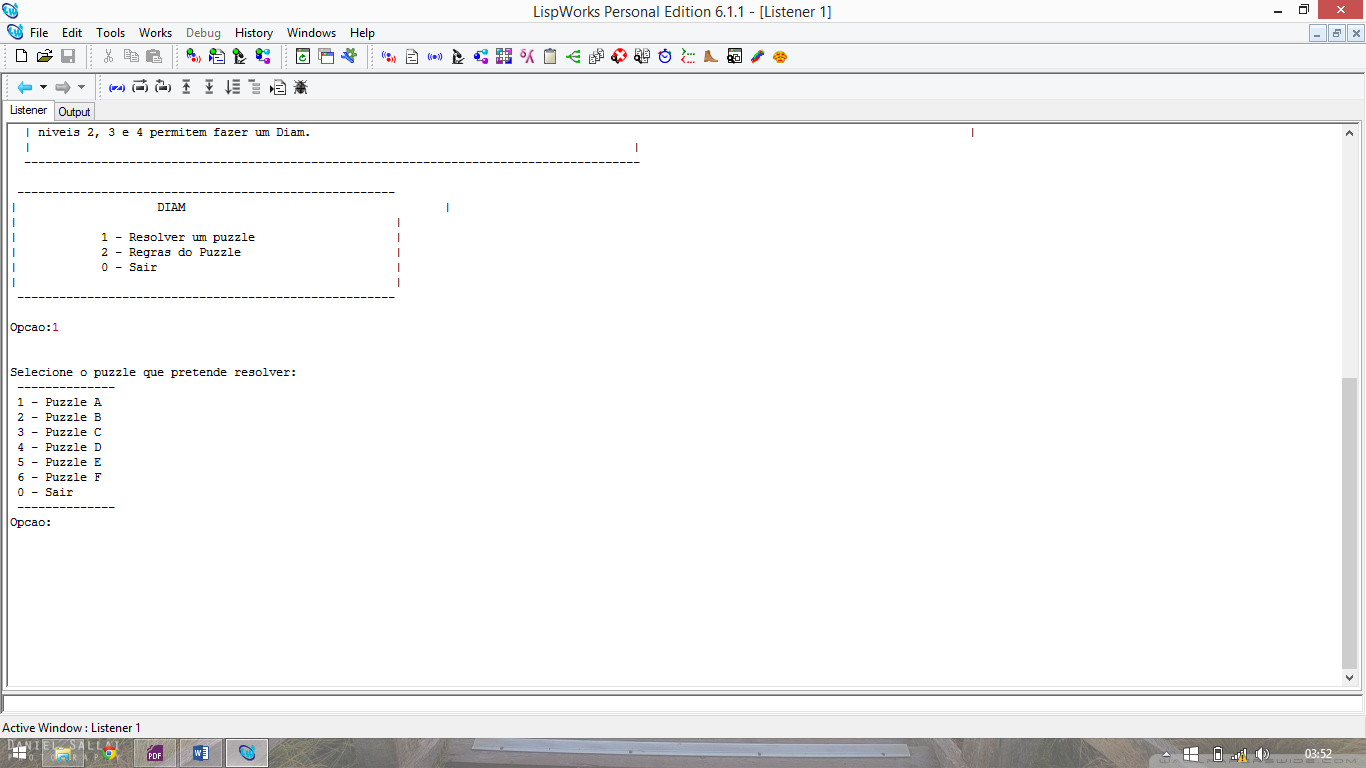


Figura 6 - Interface Selecionar Puzzle

1. Escolhe-se o puzzle que pretende resolver e este é apresentado. E é de seguida questionado sobre qual o algoritmo que pretende utilizar.

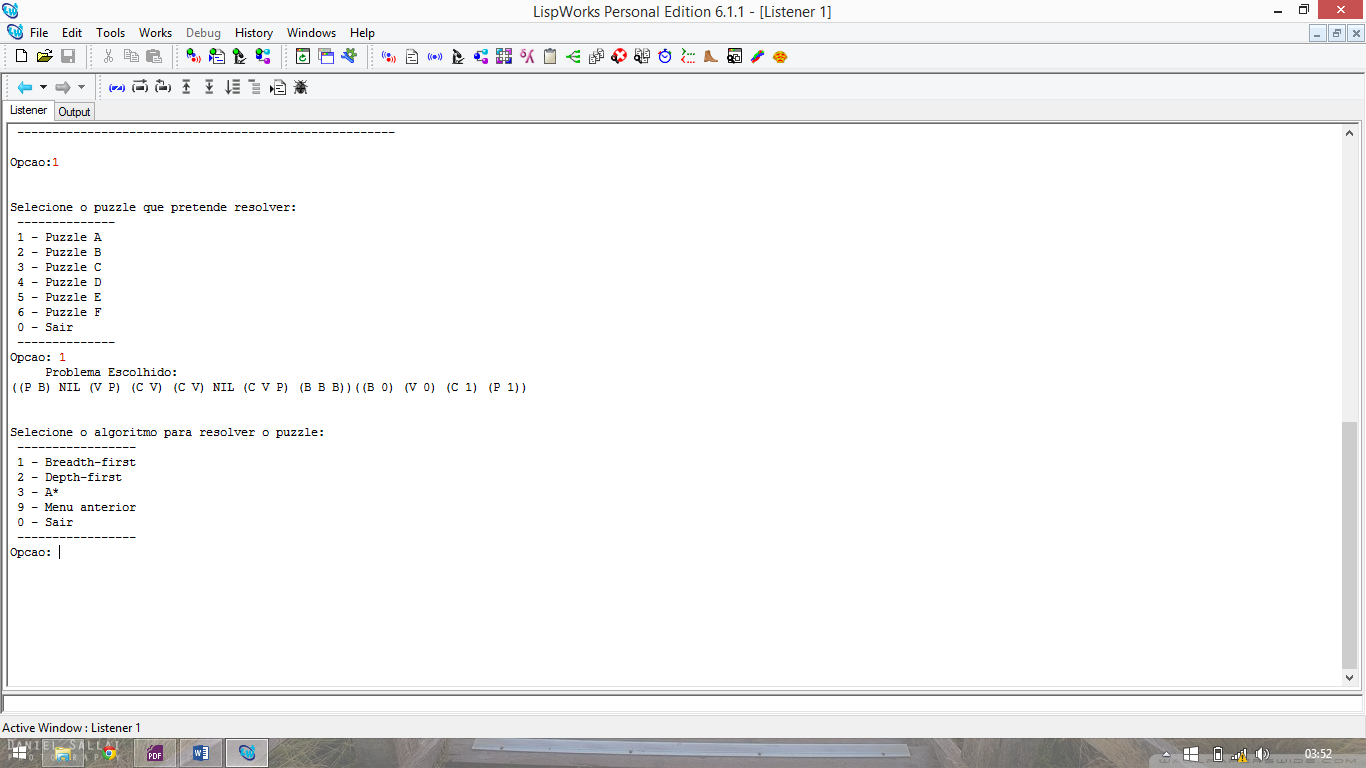


Figura 7 - Interface Escolhe Algoritmo

1. Escolhido o algoritmo, são apresentados os resultados para o algoritmo selecionado, apresentado o número de nós gerados, o número de nós expandidos, a penetrância e a profundidade da solução.

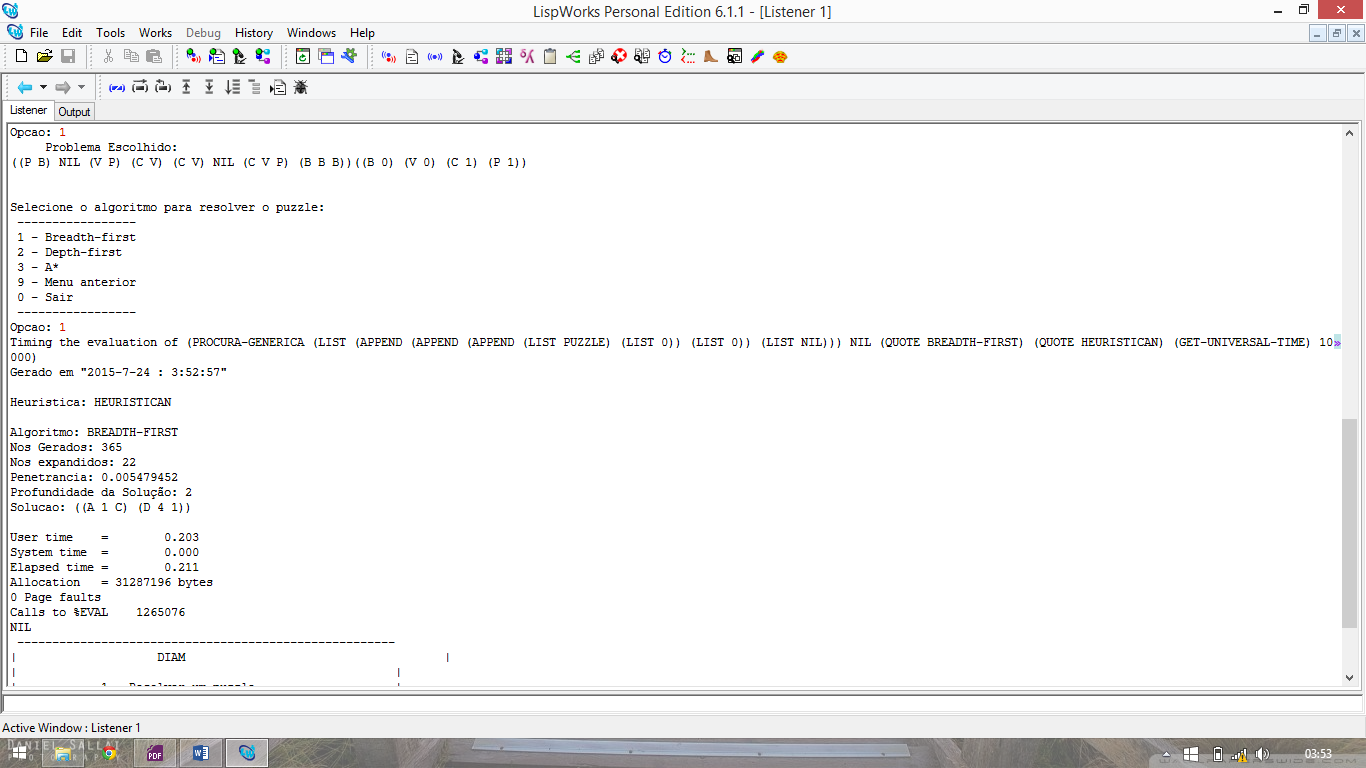


Figura 8 - Interface Resultado Algoritmo Breadth-First

1. No caso seguinte foi selecionado o algoritmo Depth-First, em que é pedido ao utilizador que insira a profundidade máxima que pretende, sendo posteriormente apresentados os resultados.

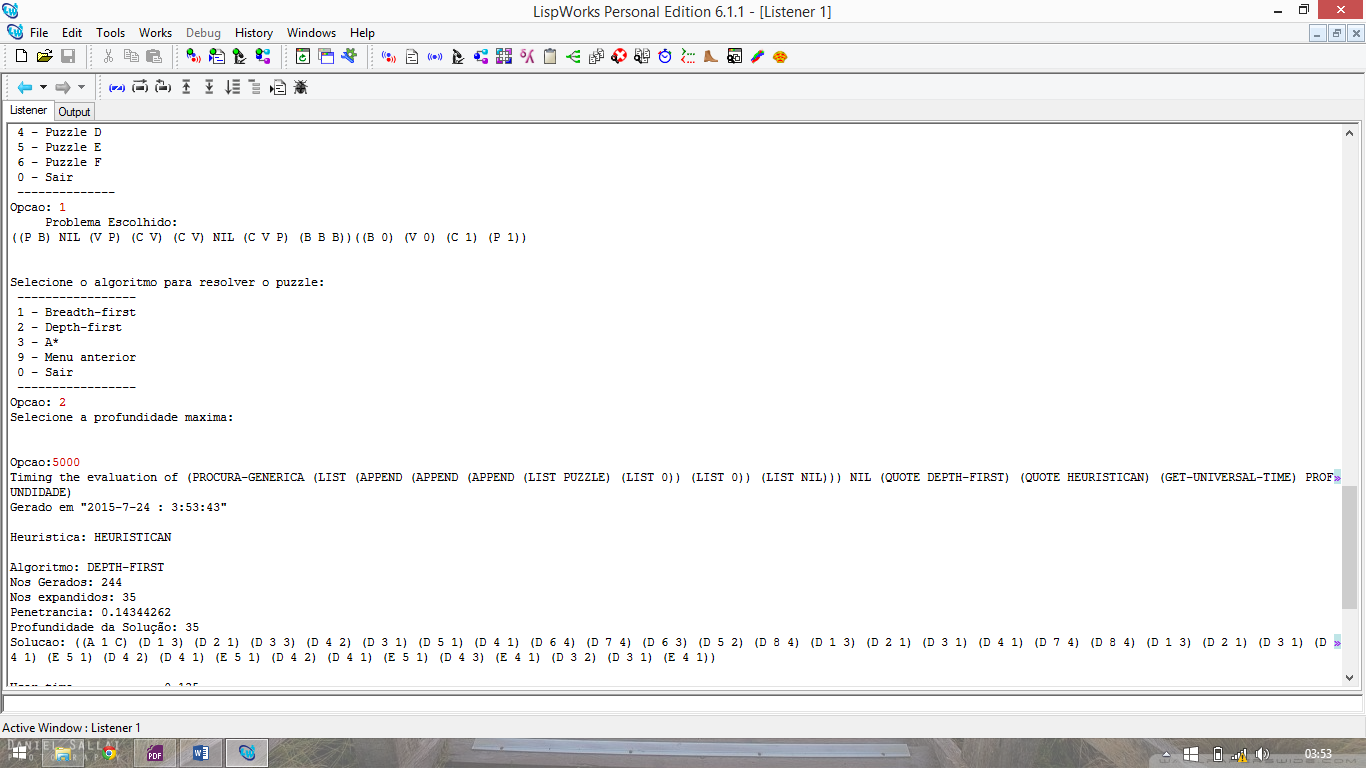


Figura 9 - Interface Resultados Algoritmo Depth-First

1. Neste caso foi selecionado o algoritmo A\* sem heurística.

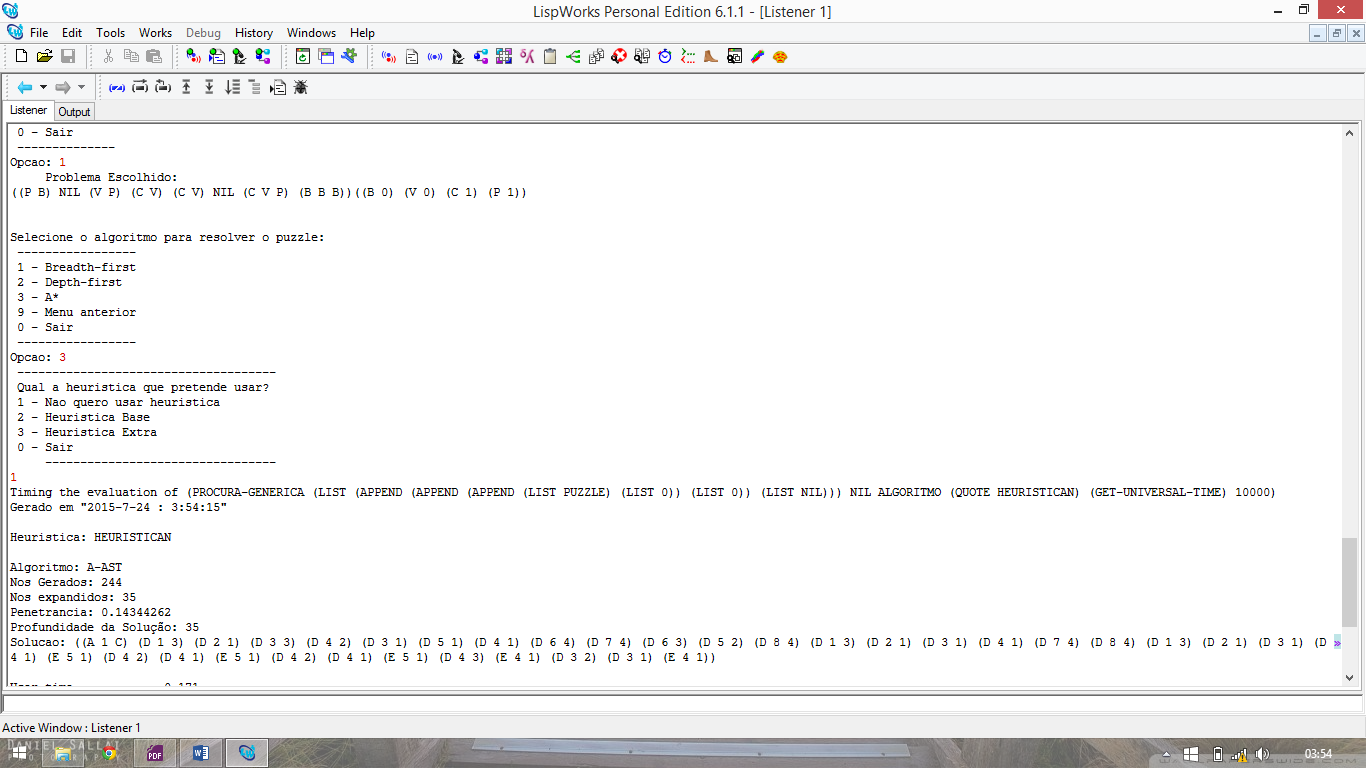


Figura 10 - Interface Resultado Algoritmo A\* Sem Heurística

1. Finalmente foi selecionado o algoritmo A\* com a heurística extra (opção 3), a título de exemplo, apesar de que as duas heurísticas funcionam.

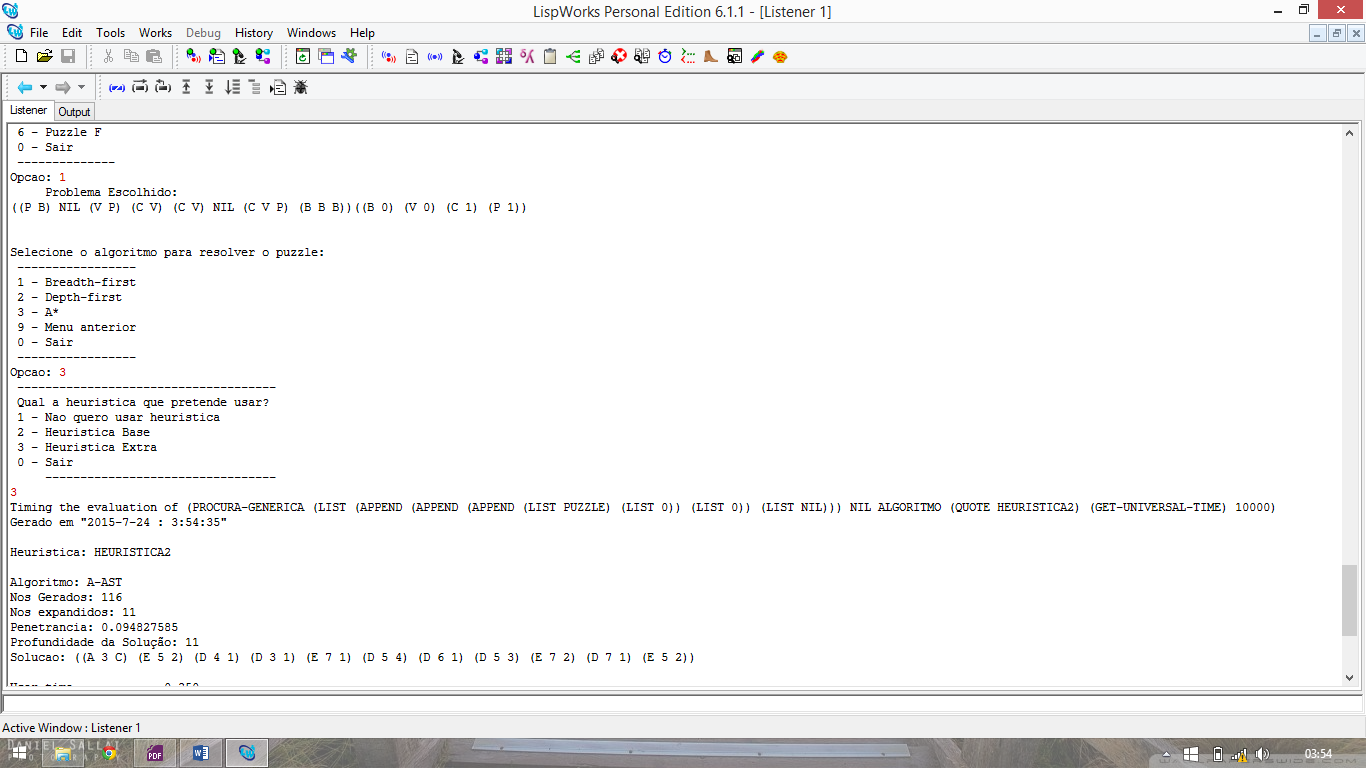


Figura 11 - Interface Resultado Algoritmo A\* c/ Heurística Extra

# Limitações da Aplicação

Não foi implementado o algoritmo de procura opcional (IDA\*, SMA ou RBFS), não por falta de conhecimento sobre o funcionamento desses algoritmos, mas por dificuldade em implementa-los concretamente.

Apresenta também uma diferença em relação ao pedido no enunciado uma vez que a lista de peças de reserva dos puzzles não é composta só pela cor das peças, mas também pela quantidade de peças que há de cada uma das cores.