

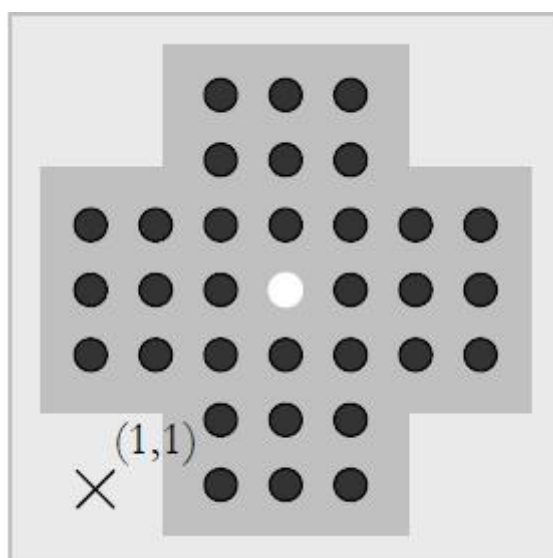
INTELIGÊNCIA ARTIFICIAL

Docentes

Prof. Joaquim Filipe

Eng. Cédric Grueau

Projecto 1 – MANUAL UTILIZADOR



“Puzzle Solitário”

Número	Nome	Turma
052206031	Zulmira Ceita	3INF01
070221158	Pedro Santos	

Setúbal

2009/2010

ÍNDICE

1	SUMÁRIO EXECUTIVO	3
2	TERMINOLOGIA E ABREVIATURAS	4
3	DESCRIÇÃO DO PROGRAMA	5
3.1	Especificações do problema.....	5
3.1.1	Início	5
3.1.2	Desenrolar do jogo	5
3.1.3	Fim de jogo	5
4	INSTALAÇÃO DO PROGRAMA	5
5	UTILIZAÇÃO DO PROGRAMA.....	6
5.1	Descrição	6
5.2	Exemplo de utilização	9
5.3	Mensagens de Erro	11

1 SUMÁRIO EXECUTIVO

O manual de utilizador é um manual concebido com o propósito de ensinar o utilizador a trabalhar com o programa que gera o jogo designado **puzzle solitário** e entender mais claramente que opções lhe são expostas facilitando, deste modo, a sua interação com o mesmo.

Contém toda a informação que o utilizador necessita para explorar o programa de modo adequado a todos os níveis.

Uma vez visto todo o manual, o utilizador deverá ser capaz executar a aplicação e poderá visualizar as estatísticas referentes ao jogo.



2 TERMINOLOGIA E ABREVIATURAS

- **Estrutura** – Conjunto de relações entre os elementos de um sistema;
- **Menu** – Lista que aparece no ecrã do computador, apresentando as opções que determinado programa ou funções permitem;
- **Menu Principal** – Primeiro menu do programa, menu raiz;
- **Opção** – Acto de escolher algo;
- **Opção Inválida** – Decisão incorrecta, inexistente, não permitida;
- **Sair** – Abandonar o programa;
- **Voltar** – Regressar ao menu anterior.
- **Algoritmo** - sequência finita de *instruções* bem definidas e não ambíguas, cada uma das quais pode ser executada mecanicamente num período de tempo finito e com uma quantidade de esforço finita.
- **Breadth - First** - algoritmo usado para realizar uma *busca* numa *estrutura em árvore* ou *grafo*. Intuitivamente, inicia-se pelo nó raiz e explora-se todos os nós vizinhos. Então, para cada um desses nós mais próximos, explora-se os nós vizinhos inexplorados e assim por diante, até que ele encontre o alvo da busca.
- **Depth - First** - algoritmo usado para realizar uma *busca* numa *estrutura em árvore* ou *grafo*. Intuitivamente, o algoritmo começa num nó raiz (seleccionando algum nó como sendo o raiz, no caso de um grafo) e explora tanto quanto possível cada um dos seus ramos, antes de retroceder.
- **A*** - algoritmo usado para realizar uma *busca* numa *estrutura em árvore* ou *grafo*. Intuitivamente, o algoritmo começa num nó raiz e possui uma função de medida de interesse que um estado tem de probabilidade de devolver a solução do problema.
- **IDA*** - algoritmo usado para realizar uma *busca* numa *estrutura em árvore* ou *grafo*. Intuitivamente, o algoritmo começa num nó raiz e possui uma função de medida de interesse que um estado tem de probabilidade de devolver a solução do problema. A expansão de um nó é limitada.
- **Heurística** - Conjunto de regras e métodos que conduzem à descoberta, à invenção e à resolução de problemas.

3 DESCRIÇÃO DO PROGRAMA

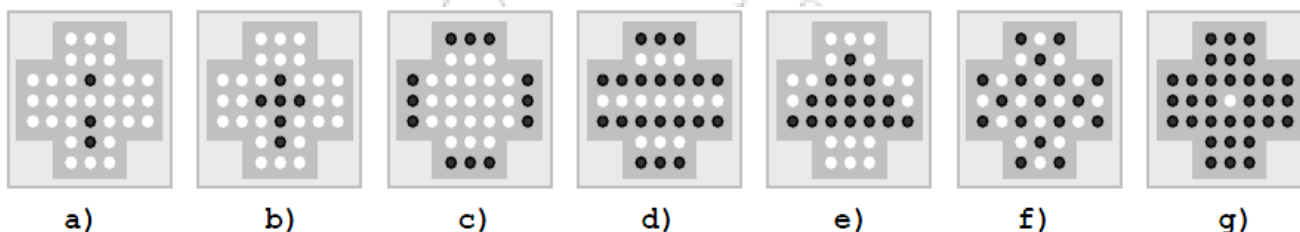
3.1 Especificações do problema

Este programa visa portanto conceder ao utilizador a hipótese de jogar o *Puzzle Solitário*. O objectivo deste projecto não consiste em resolver um jogo entre dois jogadores mas sim em otimizar os movimentos de um único jogador utilizando as regras de movimentação definidas pelo jogo. O jogo possui os seguintes componentes:

- Um jogador;
- Um tabuleiro à escolha;
- Um conjunto de regras que regem a movimentação das peças do tabuleiro.
- Um objectivo – apenas uma peça poderá ficar no tabuleiro.

3.1.1 Início

O puzzle solitário é baseado num tabuleiro representado onde as peças estão representadas por “1’s”, os espaços vazios por “0’s” e as zonas fora do tabuleiro por “nil”. A imagem seguinte mostra os tabuleiros disponíveis no jogo.



O jogador poderá escolher entre sete tabuleiros predefinidos.

O objectivo do “jogo” é ficar com apenas uma peça no tabuleiro.

3.1.2 Desenrolar do jogo

O jogo começa com a escolha do tabuleiro, do algoritmo, e das opções dependendo de cada algoritmo escolhido. Após as opções escolhidas, o jogo desenrola-se automaticamente e no fim mostra as estatísticas referentes ao jogo caso o algoritmo consiga resolver o problema.

3.1.3 Fim de jogo

O jogo termina quando apenas uma peça ficar no tabuleiro.

4 INSTALAÇÃO DO PROGRAMA

Para correr esta aplicação será necessário instalar no sistema operativo o **SDK Eclipse**.

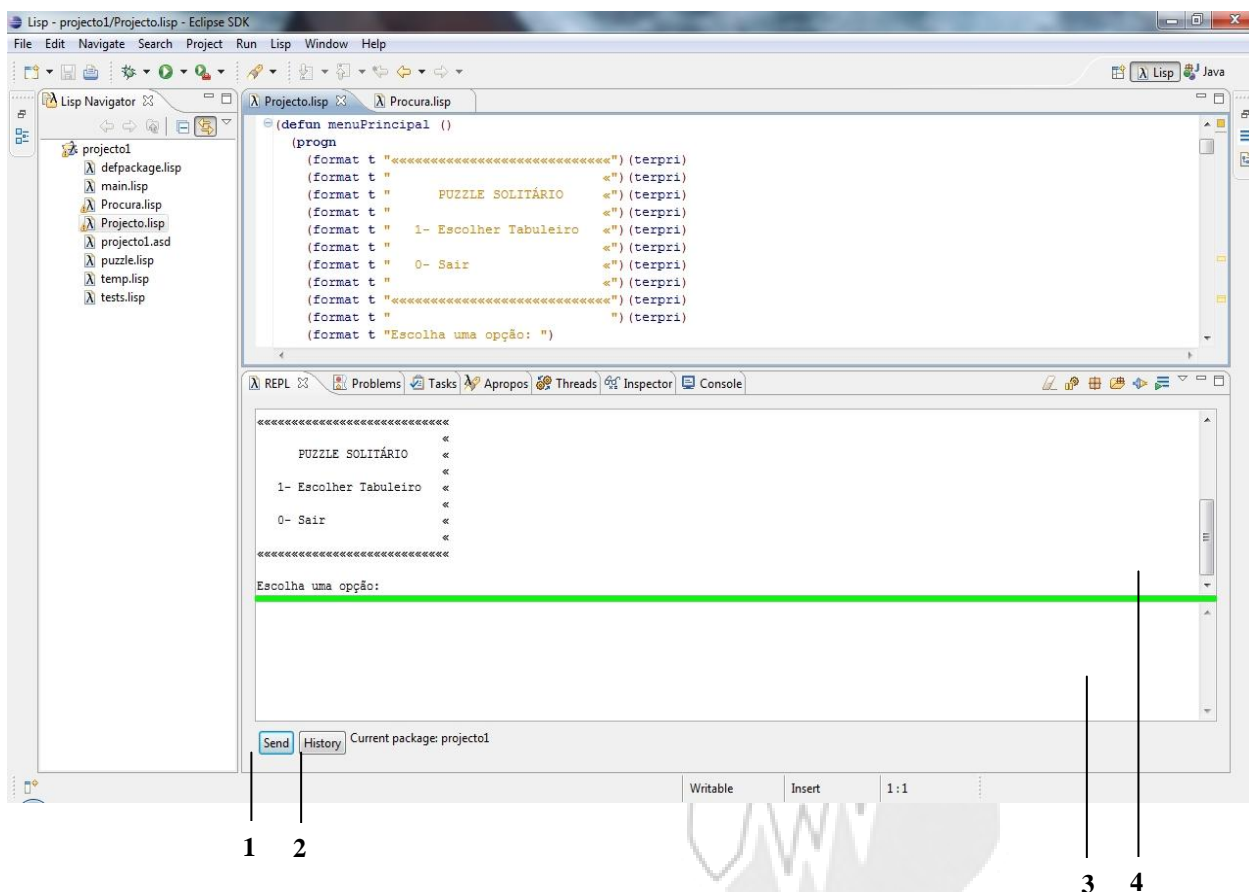
Este software está disponível efectuar o para download no site <http://www.eclipse.org/downloads/>.

Após instalado o Eclipse, deve ser instalado um **plugin** para o mesmo designado **CUSP** que está disponível para download no site <http://www.sergeykolos.com/cusp/update/>.

5 UTILIZAÇÃO DO PROGRAMA

5.1 Descrição


NOTA: Antes da utilização da aplicação, deve-se garantir que o ficheiro necessário para se ler os tabuleiros (problemas.dat) se encontra no directório 'C:\Problemas.dat'. Caso contrário deve-se proceder à sua transferência ou cópia para o respectivo directório.



1. Botão de input para submeter a opção/instrução.
2. Botão de histórico das instruções que foram submetidas.
3. Janela de input (entrada de informação).
4. Janela de output (saída de informação).

MENUS

Menu Principal



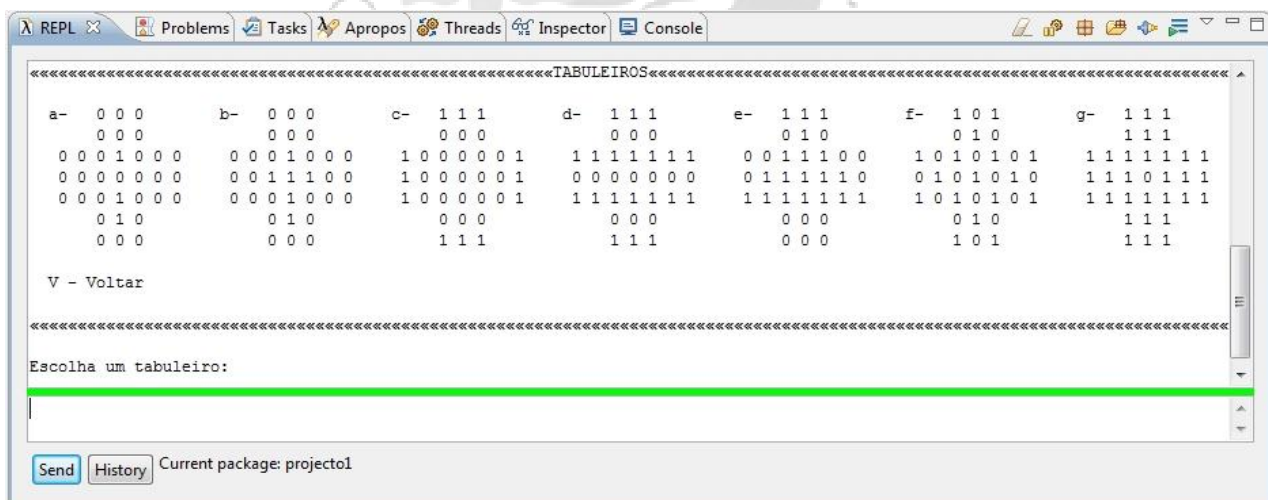
```
=====
PUZZLE SOLITÁRIO
1- Escolher Tabuleiro
0- Sair
=====

Escolha uma opção:
=====

[Send] [History] Current package: projecto1
```

- 0 - Termina o programa;
- 1 - Segue para o menu de selecção do tabuleiro;

Menu de selecção do tabuleiro



```
=====
TABULEIROS
=====

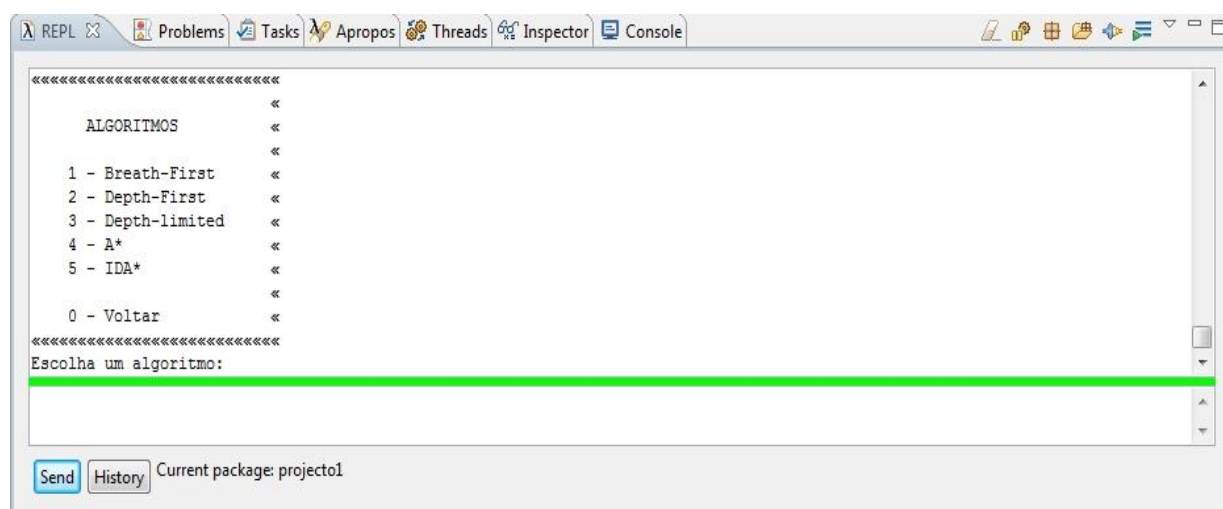
a-  0 0 0      b-  0 0 0      c-  1 1 1      d-  1 1 1      e-  1 1 1      f-  1 0 1      g-  1 1 1
    0 0 0      0 0 0      0 0 0      0 0 0      0 1 0      0 1 0      1 1 1
    0 0 0 1 0 0 0  0 0 0 1 0 0 0  1 0 0 0 0 0 1  1 1 1 1 1 1 1  0 0 1 1 1 0 0  1 0 1 0 1 0 1  1 1 1 1 1 1 1
    0 0 0 0 0 0 0  0 0 1 1 1 0 0  1 0 0 0 0 0 1  0 0 0 0 0 0 0  0 1 1 1 1 1 0  0 1 0 1 0 1 0  1 1 1 0 1 1 1
    0 0 0 1 0 0 0  0 0 0 1 0 0 0  1 0 0 0 0 0 1  1 1 1 1 1 1 1  1 1 1 1 1 1 1  1 0 1 0 1 0 1  1 1 1 1 1 1 1
    0 1 0          0 1 0          0 0 0          0 0 0          0 0 0          0 1 0          1 1 1
    0 0 0          0 0 0          1 1 1          1 1 1          0 0 0          1 0 1          1 1 1

v - Voltar

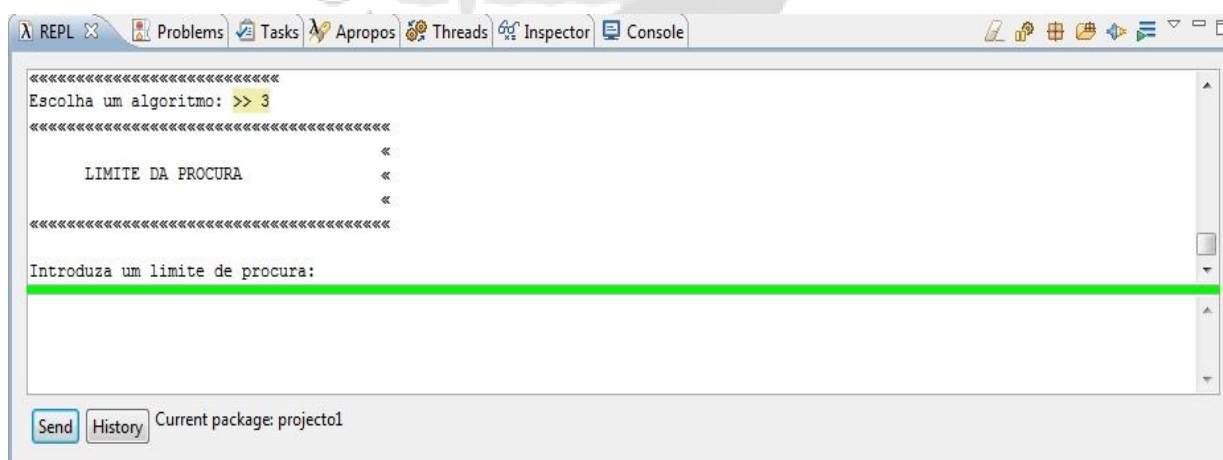
Escolha um tabuleiro:
=====

[Send] [History] Current package: projecto1
```

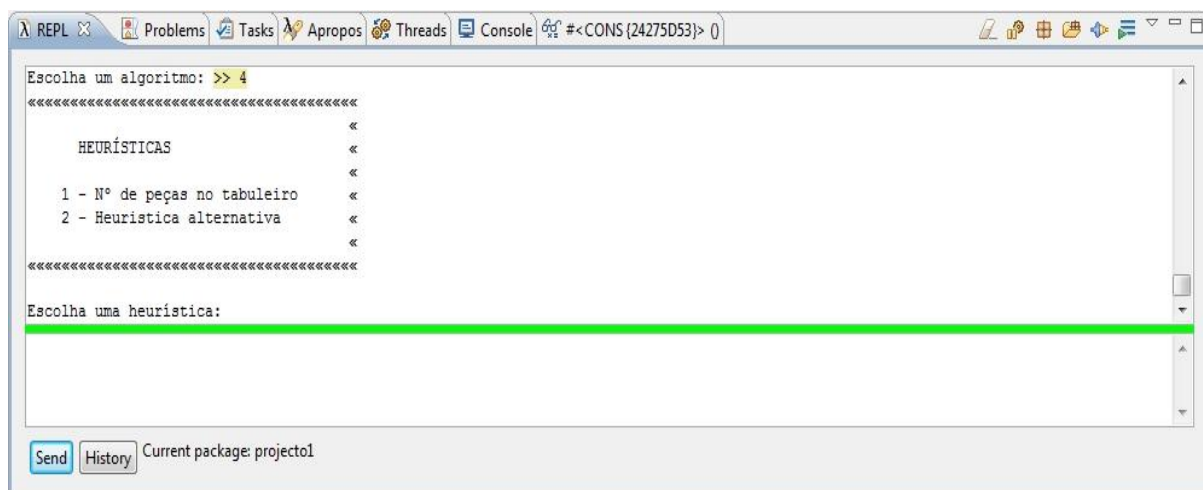
- v - Permite regressar ao menu anterior, neste caso o menu inicial;
- "a" à "g"- tabuleiros à escolha

Seleção do algoritmo que se pretende

- 1 – Procura em largura numa árvore de estados, os movimentos possíveis para o tabuleiro escolhido.
- 2 – Procura em profundidade numa árvore de estados, os movimentos possíveis para o tabuleiro escolhido.
- 3 – Procura em profundidade com um limite máximo de procura.
- 4 – Procura informada numa árvore de estados, os movimentos possíveis para o tabuleiro escolhido.
- 5 – Procura informada com iterações

Seleção do limite de procura que se pretende, no caso do algoritmo seja o Depth-Limited.

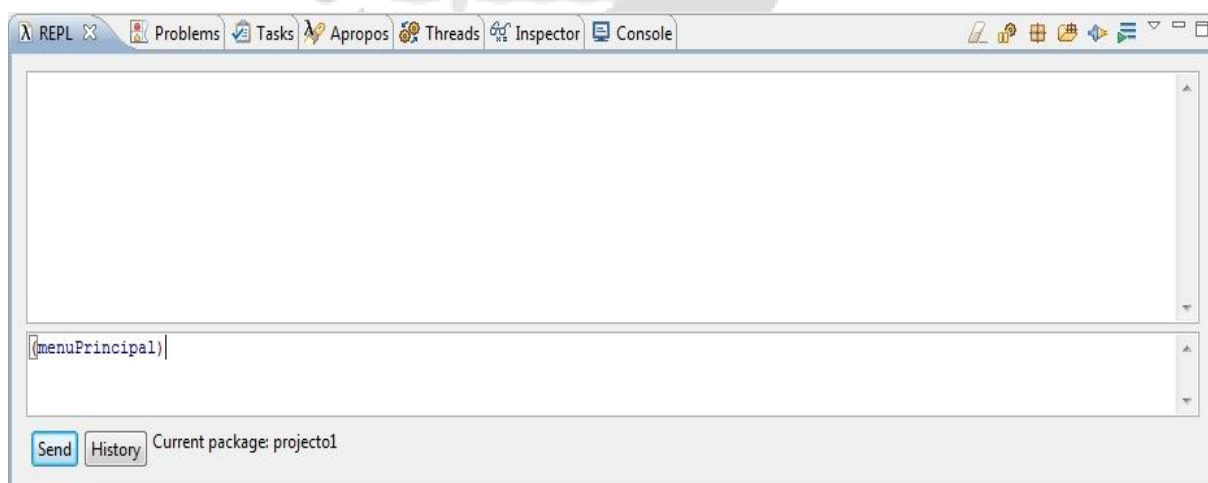
Seleção da heurística que se pretende, caso o algoritmo for o A* ou o IDA*



Após escolha do algoritmo é mostrado no ecrã todos os movimentos até a solução final.

5.2 Exemplo de utilização

1. Clique no ecrã de input.
2. Escreva com o teclado "(menuPrincipal)"



3. Prima o botão 'Send' ou prima na tecla Enter.



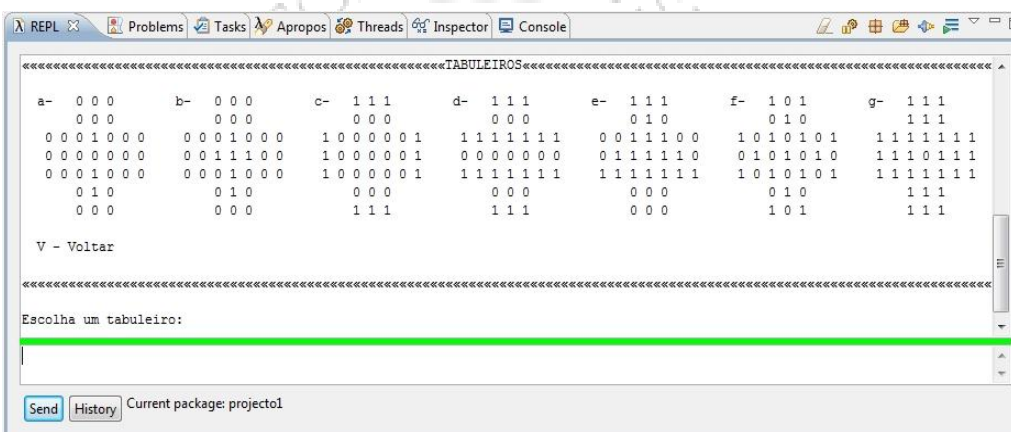
```
=====
«
PUZZLE SOLITÁRIO «
«
1- Escolher Tabuleiro «
«
0- Sair «
«
=====

Escolha uma opção:
_____

[Send] [History] Current package: project01
```

4. Prima a tecla '1'.

5. Prima o botão 'Send' ou prima na tecla Enter.



```
=====TABULEIROS=====
«
a- 0 0 0      b- 0 0 0      c- 1 1 1      d- 1 1 1      e- 1 1 1      f- 1 0 1      g- 1 1 1
   0 0 0      0 0 0      0 0 0      0 0 0      0 1 0      0 1 0      1 1 1
0 0 0 1 0 0 0 0 0 0 1 0 0 0 1 1 1 1 1 1 0 0 1 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 1 1 1 1 1 1
0 0 0 1 0 0 0 0 0 1 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1
   0 1 0      0 1 0      0 0 0      0 0 0      0 0 0      0 1 0      1 1 1
   0 0 0      0 0 0      1 1 1      1 1 1      0 0 0      1 0 1      1 1 1

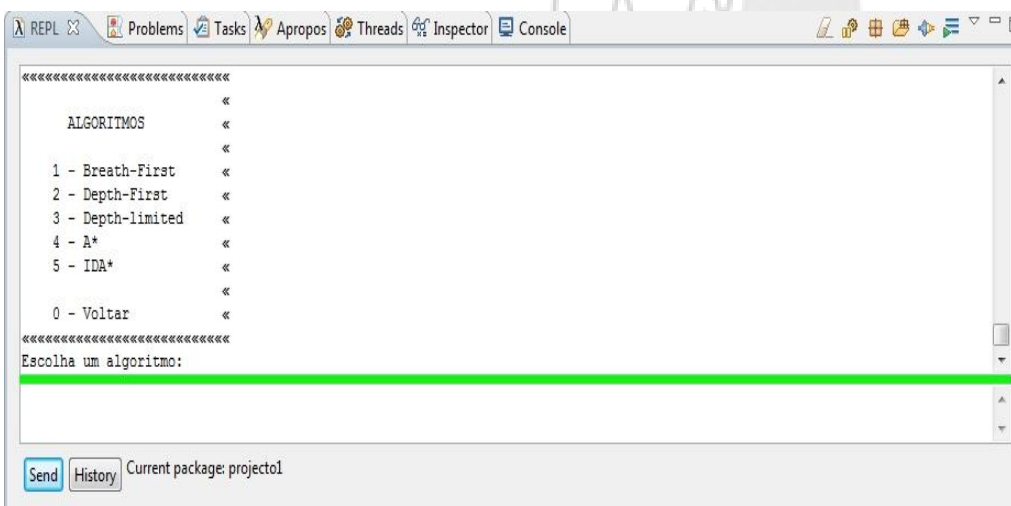
V - Voltar

Escolha um tabuleiro:
_____

[Send] [History] Current package: project01
```

6. Prima a tecla 'b' por exemplo.

7. Prima o botão 'Send' ou prima na tecla Enter



```
=====
«
ALGORITMOS «
«
1 - Breath-First «
2 - Depth-First «
3 - Depth-limited «
4 - A* «
5 - IDA* «
«
0 - Voltar «
«
=====

Escolha um algoritmo:
_____

[Send] [History] Current package: project01
```

8. Prima a tecla '4'.

9. Prima o botão 'Send' ou prima na tecla Enter.

10. Prima a tecla '1'.
11. Prima o botão 'Send' ou prima na tecla Enter.

```
Algoritmo: A-STAR
Heurística: HEURISTICA1
Limite/Limiar: NIL
Movimentos: ((CD 4 4) (CC 4 2) (CD 3 4) (CE 6 4) (CC 4 4))
Penetância: 0.031446543
Nós Gerados: 159
Nós Expandidos: 11
Profundidade da Solução: 5
Factor de ramificação médio: 31.8

Tabuleiro Inicial:
(NIL NIL 0 0 0 NIL NIL)
(NIL NIL 0 0 0 NIL NIL)
(0 0 0 1 0 0 0)
(0 0 1 1 1 0 0)
(0 0 0 1 0 0 0)
(NIL NIL 0 1 0 NIL NIL)
(NIL NIL 0 0 0 NIL NIL)

Tabuleiro Final:
(NIL NIL 0 0 0 NIL NIL)
(NIL NIL 0 1 0 NIL NIL)
(0 0 0 0 0 0 0)
(0 0 0 0 0 0 0)
(0 0 0 0 0 0 0)
(NIL NIL 0 0 0 NIL NIL)
(NIL NIL 0 0 0 NIL NIL)

Tempo decorrido:
0
Evaluation took:
0.010 seconds of real time
0.015600 seconds of total run time (0.015600 user, 0.000000 system)
160.00% CPU
24,096,348 processor cycles
716,336 bytes consed
```

12. Fim da execução do programa.
13. Vá ao directório do seu disco 'C:\solucoes.dat' com um programa de processamento de texto (exemplo: Notepad++) e veja o resultado final das jogadas com o algoritmo que escolheu no tabuleiro que seleccionou.

5.3 Mensagens de Erro

Caso a opção escolhida para cada um dos menus não seja válida, o utilizador é informado que seleccionou uma opção incorrecta e será mostrado o mesmo menu para que possa escolher a opção correcta.