

MOMENTO DE AVALIAÇÃO

Curso(s): Informática

Unidade Curricular: Inteligência Artificial

Avaliação -

Docente: Ricardo Malheiro

Duração: -

Data: -

SUDOKU Solver

Sudoku, por vezes escrito Su Doku, é um quebra-cabeças baseado na colocação lógica de números. O nome deste jogo deve-se ao facto de, em japonês, "su" querer dizer "número" e "doku" significar "único". Este nome está diretamente relacionado com o objetivo do quebra-cabeças: colocação de números de 1 a 9, em cada uma das células vazias no tabuleiro, de forma a que não existam números repetidos por linha, por coluna ou por sub-matriz. O tabuleiro de jogo consiste numa grelha ou matriz de 9×9 células, sendo também composta por 3×3 sub-matrizes, tendo cada uma 3×3 células, como se pode ver na Figura 1.

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Figura 1 - Exemplo de um Jogo de Sudoku

Normalmente cada Sudoku tem apenas uma única solução. A solução do Sudoku apresentado na Figura 1 é representada na Figura 2, com as respetivas células marcadas a vermelho.

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

Figura 2 - Solução do Sudoku inicialmente apresentado

É fornecido aos alunos a infra-estrutura base de código em java que os alunos deverão utilizar para fazer o projeto. Essa infra-estrutura para estar totalmente funcional tem em falta o código do método *executar()* da classe *AlgoritmoGenetico*.

Neste código, nenhuma das 81 posições está fixa, i.e., o algoritmo gera inicialmente números de 1 a 9 para todas as 81 posições.

Tarefas a implementar no projeto:

a) Implementar o método *executar()* da classe *AlgoritmoGenetico*.

b) Implementar o método de seleção *Roleta*.

c) Implementar o método de recombinação com 2 pontos de corte.

d) Implementar o método de recombinação uniforme.

e) Adaptar o código para poder resolver um qualquer puzzle (com solução) fornecido. Neste caso o algoritmo deve ser testado para o seguinte puzzle de entrada:

5	3			7		
6			1	9	5	
	9	8				6
-----+-----+-----						
8				6		3
4			8		3	1
7				2		6
-----+-----+-----						
	6				2	8
			4	1	9	5
				8		7
						9

O programa deverá gerar o seguinte ficheiro com a solução encontrada para o problema:

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
-----+-----+-----								
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
-----+-----+-----								
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

f) O sistema deve ler o(s) puzzle(s) a partir de ficheiro(s) .txt. Esses ficheiros devem ter o nome puzzle#.txt e puzzle#_solved.txt. Serão valorizadas soluções em que a manipulação dos ficheiros seja efetuada pelo interface gráfico já fornecido.

Material a entregar:

Relatório, onde deverá descrever detalhadamente:

- A Codificação dos Indivíduos
- Os Operadores Utilizados:
 - Operadores de Seleção
 - Operadores de Recombinação
 - Operadores de Mutação
- A Função de Fitness

Avaliação

Os parâmetros de avaliação são os seguintes:

Método executar()	8%
Método de seleção Roleta	15%
Método de recombinação com 2 pontos de corte	15%
Método de recombinação uniforme	15%
Resolver qualquer tipo de puzzle	20%
Leitura e escrita de ficheiros em consola (cotado para apenas 50%), em ambiente gráfico (cotada para 100%)	10%
Defesa	12%
Relatório (Restante)	5%
Total	100%

Notas Finais

1. O trabalho pode ser realizado individualmente ou em grupos de, no máximo, 2 alunos.
2. Nos elementos entregues (código e relatório), devem ser mencionados obrigatoriamente o número e o nome completo dos alunos.
3. Qualquer sinal de cópia/plágio implicará a anulação do trabalho.