

Faculdade de Ciências e Tecnologias – Universidade
do Algarve

Licenciatura em Engenharia Informática

Unidade Curricular: Inteligência Artificial



Problema 4: N-queens resolvido com um algoritmo genético

Ano letivo: 2022/23

Semestre: 1º

Grupo No: 3

Turma: PL1

Trabalho realizado por:

Nome	Número
Daniel Ferros Fernandes	71320
Diogo Afonso Nobre Zacarias	71323
Kartic Hitendra Premgi	71379

Descrição do problema:

Este trabalho consiste em escrever um programa capaz de resolver o problema n-queens, utilizando um algoritmo genético, o problema é considerado resolvido se o programa achar uma solução dentro do melhor tempo possível.

O problema n-queens consiste em colocar n rainhas em um tabuleiro de xadrez $n \times n$, sem que nenhuma das rainhas se ataquem. Uma vez que segundo as regras de xadrez, a rainha anda por todo o tabuleiro na horizontal, vertical e nas suas respectivas diagonais.

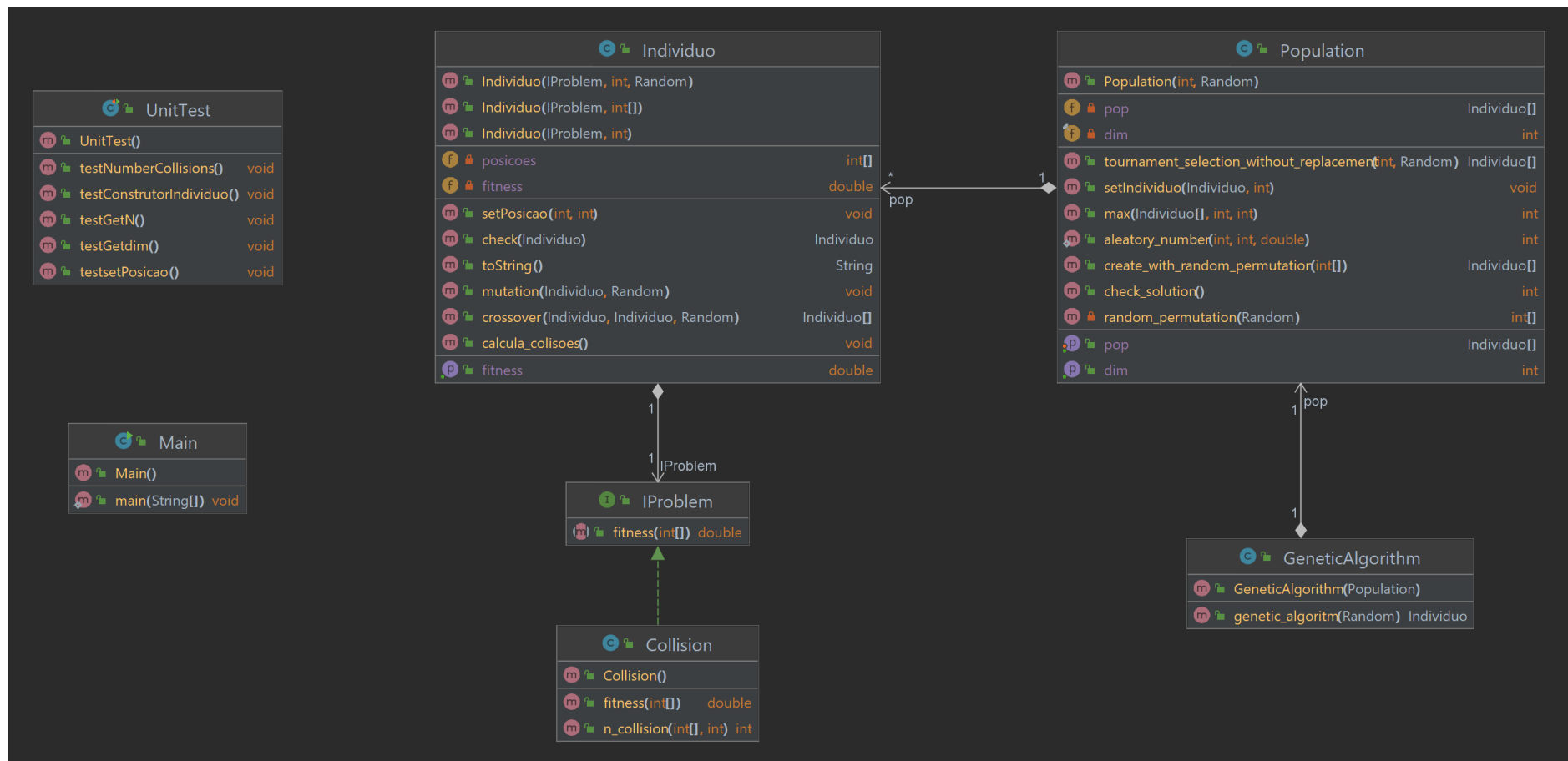
Algoritmo Genético:

Um algoritmo genético é um tipo de algoritmo que busca encontrar soluções para problemas complexos de otimização, inspirado no processo de evolução natural das espécies.

O funcionamento do algoritmo genético envolve os seguintes passos:

1. **Geração da população inicial:** o algoritmo gera uma população de soluções possíveis aleatoriamente. Cada solução é chamada de indivíduo, e é representada por um board com n rainhas e tem um fitness, que representa o número de rainhas livres;
2. **Seleção dos indivíduos:** os indivíduos são selecionados para reprodução de acordo com a sua aptidão. Os indivíduos mais aptos têm mais chances de serem selecionados para reprodução, ou seja, aqueles que têm rainhas mais livres são os mais prováveis a ser selecionados;
3. **Reprodução dos indivíduos:** os indivíduos selecionados para reprodução são combinados entre si através de operações genéticas, como crossover e mutação. O crossover é o processo de combinação de características de dois indivíduos para criar dois indivíduos. A mutação é o processo de alteração aleatória de uma das rainhas de um indivíduo;
4. **Geração de nova população:** a nova população é formada pelos indivíduos resultantes da reprodução e pelos indivíduos mais aptos da população anterior, isto significa, a atualização da população;
5. **Repetição dos passos 2 a 4:** até que seja alcançada uma solução, isto é, encontrar n rainhas livres;

Diagrama de Classes UML:



Todas as opções de projeto tomadas, designadamente padrões de projeto

▪ Torneio de seleção sem reposição:

O torneio de seleção é um método usado para selecionar indivíduos de uma população através de um torneio onde se realiza a comparação de fitness para determinar os indivíduos mais aptos para reprodução.

Como a dimensão da nossa população é sempre 100 elementos depois de testes verificamos que uma boa dimensão para o torneio seriam 20 elementos.

Para a realização do torneio começamos por criar um array com posições randomizadas depois preenchemos um array de indivíduos com essa mesma ordem em seguida fazemos comparações de fitness de cada elemento com os 20 elementos seguintes do torneio realizando assim 5 iterações totalizando 5 vencedores.

Fazemos isto 20 vezes totalizando assim 100 elementos vencedores e formando uma nova população.

▪ Crossover:

O crossover é a criação de dois filhos através de dois pais. A ideia que nós utilizamos foi selecionar dois índices aleatórios, e trocar esse intervalo nos dois filhos. E, depois, encher o resto do array com os números que faltam. Exemplo:

Pai 1: [1, 2, 0, 3];

Pai 2: [0, 3, 2, 1];

E assim os filhos ficaram assim, se o intervalo fosse 1 a 2:

Filho 1: [0, 3, 1, 2];

Filho 2: [0, 2, 1, 3];

▪ Mutação:

A mutação em um algoritmo genético é uma operação que ocorre aleatoriamente em um indivíduo da população com o objetivo de introduzir novas soluções e promover a diversidade genética. No caso do nosso problema, a estratégia de mutação seria escolher dois números aleatórios da sequência e trocá-los de posição. Isso pode gerar uma nova solução válida e pode ser útil para evitar que a população fique presa em um mínimo local.

É importante notar que a mutação deve ser realizada de forma controlada, pois se ela ocorrer com muita frequência pode levar a soluções inválidas ou a uma perda de informação útil. Por isso, a taxa de mutação deve ser ajustada, assim a taxa que nós escolhemos foi de 0,8, pois se a taxa de mutação for alta, existe menor possibilidade de uma dada posição ficar estagnada, ou seja, é possível a busca tornar-se mais aleatória.

Conclusão/Observações:

A resolução do problema das N queens através de algoritmos genéticos tem se mostrado uma abordagem eficiente e viável. Neste relatório, foi apresentado um algoritmo genético específico para resolver o problema das N queens, explicando cada etapa do processo de forma detalhada.

Os resultados obtidos foram satisfatórios, com a maioria das execuções encontrando uma solução viável em um tempo razoável. Além disso, o algoritmo genético apresentou uma alta taxa de convergência, o que indica que ele é capaz de encontrar rapidamente uma solução viável para o problema.

Alguns resultados obtidos foram, por exemplo, com a solução do problema ser encontrada em uma média de 3 gerações para tabuleiros de tamanho 8x8, num tempo de execução de 0,007 segundos e em uma média de 207 gerações para tabuleiros de tamanho 16x16, num tempo de execução de 0,065 segundos.

Em resumo, o algoritmo genético apresentado neste relatório mostrou-se uma opção eficaz para resolver o problema das N queens, com resultados consistentes. Dessa forma, é possível concluir que a utilização de algoritmos genéticos pode ser uma opção viável para resolver problemas de otimização em geral.

Referências bibliográficas:

https://rstudio-pubs-static.s3.amazonaws.com/257269_c649348b63c94bcf8068249f594840be.html