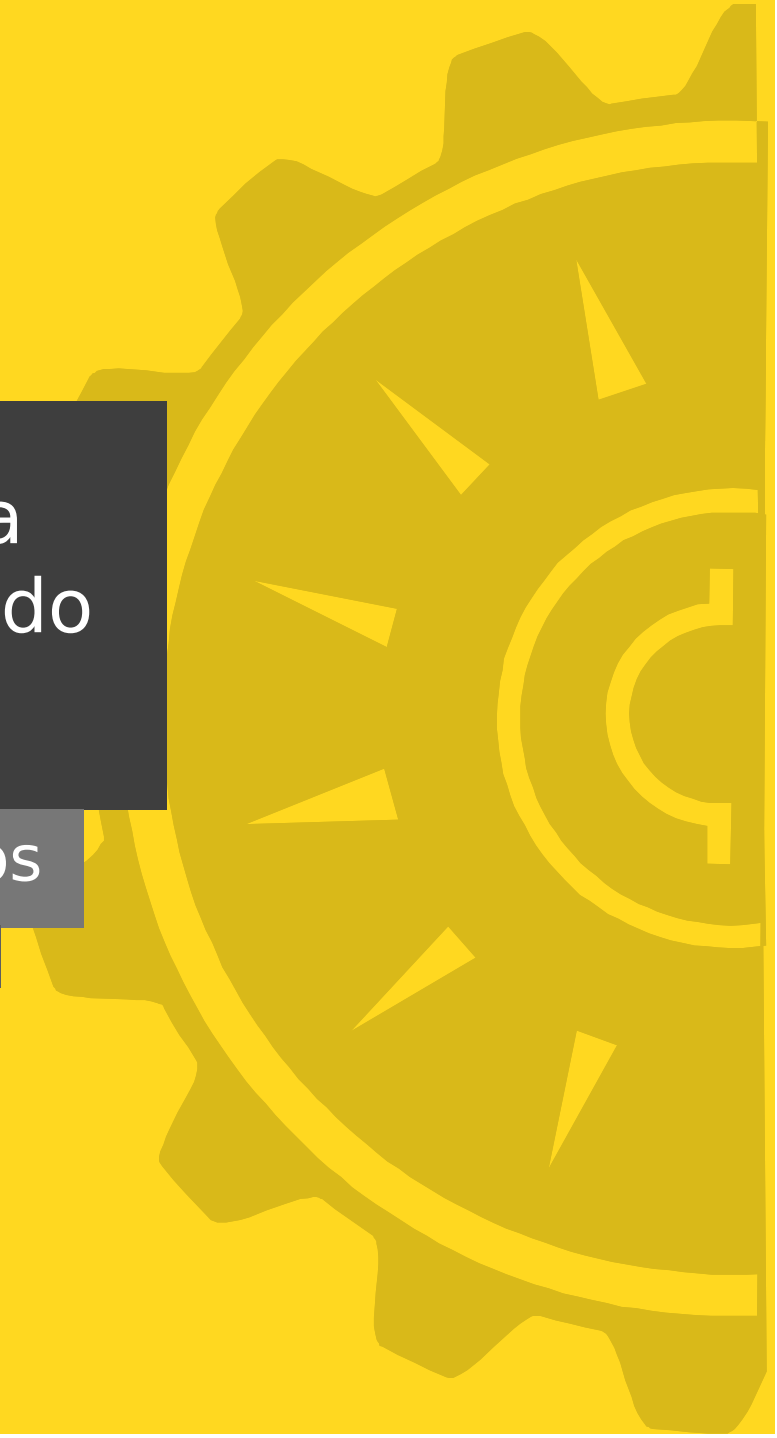
Estudo do funcionamento e da complexidade computacional do algoritmo A*

Estruturas de Dados e Algoritmos

11907 - Marco Sacristão



Objectivos e Motivação

Este trabalho prático tem como objetivo a programação do algoritmo A* a ser utilizado, na linguagem de programação Java, sendo esta a linguagem de programação escolhida é também necessário a implementação básica de uma GUI de maneira a demonstrar os resultados obtidos sobre a imagem digital a ser lida em conjunto com a biblioteca OpenCV.

Após a programação do algoritmo A* e demonstração do seu correto funcionamento, deverá ser feita a exportação do tempo que o mesmo demorou a encontrar o melhor caminho possível, analisando assim os resultados obtidos.

Contributos

O trabalho prático foi apenas desenvolvido por um elemento como descrito na capa do mesmo, sendo assim não existiu divisão de tarefas para resolução do mesmo, logo, todo o produto resultante é de mérito do autor apenas.

PGM parsing

O formato PGM permite-nos saber através do seu cabeçalho (por esta ordem de leitura em linhas):

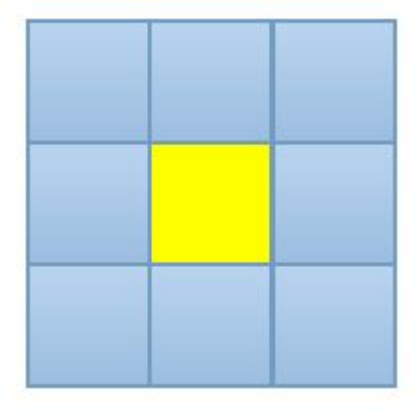
- **Magic number:** Permite-nos saber qual o tipo de ficheiro que estamos a lidar, o magic number de uma imagem com o formato PGM é “P5”.
- **Altura e Largura:** Diz-nos através do formato de dois números inteiros separados por um espaço qual a altura e largura da imagem.
- **Valor Máximo:** Qual o valor máximo possível que um pixel pode assumir em cor, em que o valor mais próximo de zero será branco, e o valor mais próximo do valor máximo apresentado será preto (pois trata-se de uma imagem em Grayscale).
- **Valores:** As restantes linhas do ficheiro contém então um valor de zero até ao valor máximo, e existem portanto Altura× Largura valores contidos até ao fim do ficheiro.

Conversão PGM para PNG

Não existindo suporte para nativo para Java no que toca a ficheiro PGM é necessário recorrer à biblioteca OpenCV e utilizar a mesma para gravar a imagem em PNG de modo a ser possível utilizar a imagem e apresenta-la na janela. A leitura da cor do pixel apesar de ser feita na matriz realizada pelo ficheiro parsing do ficheiro PGM, o mesmo não dispõe de uma maneira que seja possível ler o pixel numa determinada coordenada x,y portanto, teremos de utilizar novamente o formato PGM (usando a biblioteca OpenCV) para ler pixel a pixel depois de recebida a lista que contém os pixeis (descritos por coordenadas em x,y).

Algoritmo A*

O A* é um algoritmo de pathfinding, permite-nos calcular o melhor caminho entre dois pontos (evitando eventuais paredes ou obstáculos), neste caso, tendo em conta o valor da cor, o mesmo procura vizinhos que rodeiam o pixel num valor desse ponto e x-1 até x+1, e y-1 até y+1, como demonstrado na imagem abaixo (pixel atual representado pela cor amarela, sendo os azuis a vizinhança do mesmo).



A formula base para o calculo é (F = G + H) em que H representa distância do ponto inicial até ao ponto final.

Dispositivos utilizados

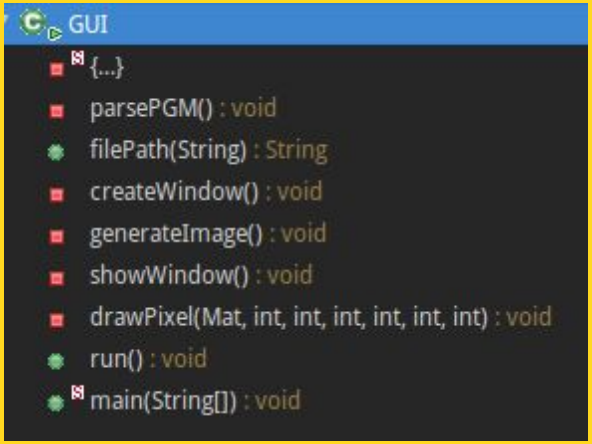
A compilação deste projeto em Java teve como alvo uma máquina com sistema operativo Linux tendo sido gerido o mesmo no IDE Eclipse.

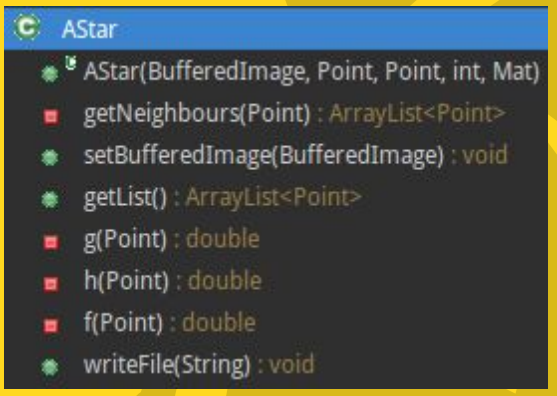
As características da máquina são as seguintes:

- **CPU:** Intel Core i7 3537U
- **RAM:** 8GB
- **Disco:** Crucial m4 128GB (6Gb/s)

Sistema Experimental

O sistema experimental em Java está divido em duas classes “GUI.java” e “AStar.java”. Esta é a divisão mais simples possível tendo em conta o paradigma da orientação a objetos. A classe GUI é responsável pela criação e apresentação da janela, assim como de geração das imagens necessárias

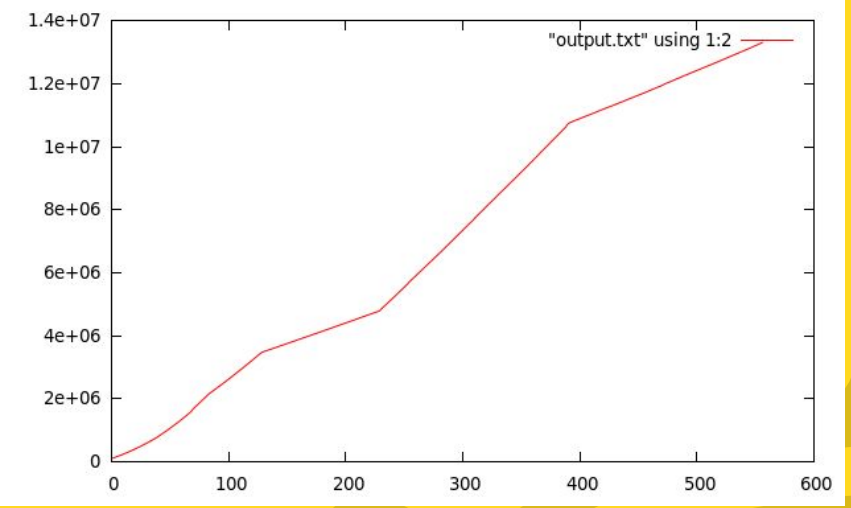




Análise dos resultados

De modo a obter resultados, foi calculado o tempo total da procura de todos os pontos no decorrer do pathfinding do nosso algoritmo.

Na totalidade são percorridos 516 pontos, é nos possível saber qual o tempo que demorou em cada ponto em nano-segundos. Analisando este tempo, podemos verificar que se trata de um crescimento linear.



O cálculo de cada ponto (Do algoritmo A* em si) pode ser definido com complexidade computacional de O(n). com complexidade O(n). Visto que o tempo de cálculo do mesmo não têm influência dos anteriores.

Conclusão

Foi realizada pesquisa heurística por grafos A*, numa imagem de formato PGM.

A utilização desse mesmo algoritmo deve ser efetuada sobre uma imagem, de forma a encontrar o melhor caminho entre dois pontos nessa imagem, através do seu custo em distância, e utilizando a cor mais próxima entre os pontos iterados. A leitura da imagem é efetuada de modo a saber o seu tamanho e conteúdos, utilizando a biblioteca OpenCV, a qual permite a leitura da imagem em formato PGM permitindo a sua transformação numa imagem de outro formato de modo a saber o valor da cor em cada pixel.

É então que analisámos os resultados obtidos, podemos verificar que o tempo de execução do algoritmo em si é um crescimento linear, pois o crescimento dá-se pela soma do tempo de pesquisa de cada ponto, de todos os pontos necessário, desde o momento em que o algoritmo se inicia, até que o mesmo termina.

O algoritmo A* apresenta uma complexidade computacional de O(n).