T1 Algoritmos e Estrutura de Dados II

Guilherme Velho Wojtysiak

PUCRS

Resumo. *A polícia precisou perseguir uma quadrilha que recém havia roubado um banco, durante a perseguição já aproveitou para recolher o dinheiro que foi caindo. Agora eles apresentam um mapa para a equipe de perícia, informado a rota de fuga dos bandidos e em quais locais foram recolhidas notas junto com o seu valor total. A equipe de perícia pediu a minha ajuda para analisar os mapas e contabilizar quanto dinheiro foi recolhido ao final da perseguição.*

1. Introdução

Desenvolver um software que realiza a leitura de um mapa e calcula a quantia recuperada durante a perseguição da polícia.

2. O problema

Para estruturar o mapa, foram utilizados símbolos (-,/,\,| e #) para determinar a direção em que a perseguição seguiu, sendo "-" utilizado para informar movimento Horizontal, "/" e "\" para informar transição do Horizontal para Vertical ou Vertical para Horizontal, "|" para informar movimento Vertical e "#" para informar o fim da perseguição. Para informar onde e quanto dinheiro foi recuperado, o mapa também inclui valores entre os símbolos (ex: --3--2--322--).

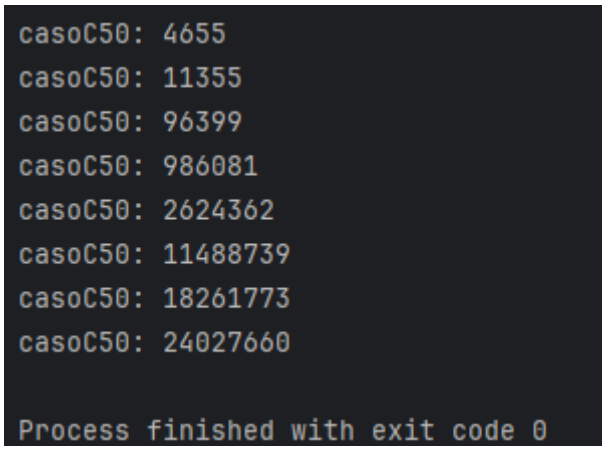
3. Processo de solucao

Para a solução do problema, foi utilizada a API Java Buffered Reader para a leitura do mapa. Cada linha foi processada individualmente, sendo cada um de seus caracteres transferidos para uma matriz X por Y (sendo X o tamanho do mapa na Horizontal e Y na Vertical) através do método String.charAt(). Após a transferência, foi utilizado um algoritmo para encontrar o início do mapa na primeira coluna da matriz, representado por um caractere "-". Para a interpretação do mapa, foi implementado um switch case com instruções sobre como tratar a ocorrência de cada um dos símbolos, utilizando variáveis como coordenada atual e direção (+-X ou +-Y) que o software está percorrendo.

4. Evidências de que o problema foi resolvido

Comparativo do resultado recolhido da solução com o disponibilizado pelo repositório do GitHub:

	Solução	GitHub
casoC50	4655	4655
casoC100	11355	11355
casoC200	96399	96398
casoC500	986081	986081
casoC750	2624362	2624362
casoC1000	11488739	11488739
casoC1500	18261773	18261773
casoC2000	24027660	24027660



(captura de tela do IntelliJ)

5. Conclusões

A maior dificuldade que encontrei durante o desenvolvimento da solução foi a criação do algoritmo para executar as instrução de mudança de direção (“/” e “\”). Como aprendizado, percebi que a atividade se assemelha muito com um compilador, logo, sinto que ela serviu como exercício para desenvolver um futuramente.