

Faculdade de Engenharia da Universidade do Porto



Universidade do Porto

FEUP Faculdade de Engenharia

Proteção Civil

Algoritmos e Estruturas de Dados

Novembro 2017 - MIEIC

Equipa: Turma1_G7

Elementos:

Helena Montenegro: up201604184@fe.up.pt

Mariana Dias: up201606486@fe.up.pt

Índice

1. Introdução.....	3
2. Sistema.....	4
2.1. Postos de Socorro.....	4
2.2. Acidentes.....	5
2.3. Tratamento de Acidentes.....	6
2.4. Visualização de Informação.....	7
2.5. Guardar Informação Alterada.....	7
3. Diagrama de Classes.....	8
4. Principais dificuldades.....	9
5. Contribuição de cada elemento do grupo.....	9

1. Introdução

No âmbito da disciplina de Algoritmos e Estruturas de Dados, desenvolvemos um projeto com o tema “Proteção Civil”, que consiste na criação de um sistema que permita atribuir socorristas aos acidentes que ocorrem em Portugal, de uma forma eficiente.

2. Sistema

O sistema desenvolvido é constituído pelos acidentes e pelos postos de socorro existentes para atender aos acidentes. Tanto os acidentes como os postos são submetidos na base de dados, no início do programa, através de ficheiros de texto, de nome “acidentes.txt” para os primeiros, e “postos.txt” para os segundos, e, no decorrer do programa, através das opções B e C do menu inicial. A classe Sistema, que é a classe principal e onde foram feitas as funções que permitem fazer a atribuição de socorristas a acidentes, é constituída por um vetor de apontadores para acidentes e por um vetor de apontadores para postos de socorro.

2.1. Postos de Socorro

Os elementos do vetor postos, atributo da classe Sistema, são objetos da classe base PostoSocorro. PostoSocorro tem como atributos o local onde se situa o posto, do tipo classe Local, sendo constituída pelo seu distrito e código postal, o número de socorristas disponíveis no posto, o número de veículos existentes no posto e um vetor de acidentes da classe Acidente que representa o histórico de acidentes que o dado posto socorreu.

Os postos de socorro podem ser de três tipos: bombeiros, polícia e INEM.

O posto de socorro dos bombeiros tem como atributos o número de ambulância e o número de autotanques, enquanto que a polícia tem como atributos valores booleanos que correspondem ao tipo de veículo que utiliza, motos ou carros. O INEM, à semelhança da polícia, também tem como atributos valores booleanos que caracterizam o tipo de veículo que o posto utiliza— motos, carros ou ambulâncias — sendo que uma ambulância é o veículo que demora mais tempo e tem como capacidade total dois socorristas, um carro é relativamente mais rápido e tem a mesma capacidade que a ambulância e a moto é a mais rápida e leva um socorrista.

Através da opção C do menu, pode-se adicionar um novo posto de socorro à base de dados. É possível editar um posto de socorro com a opção D.

2.2. Acidentes

Os elementos do vetor de acidentes, presente na classe sistema, são objetos da classe base Acidente. Esta classe tem como elementos uma data na qual ocorreu o acidente, que pertence à classe Data, constituída por um dia, um mês e um ano; o local onde ocorreu o acidente, pertencente à classe Local, que é composta por um distrito e um código postal; e uma variável do tipo bool, denominada “tratado”, que indica se o acidente já foi socorrido.

Os acidentes podem ser de três tipos: incêndios, assaltos ou acidentes de viação.

Os incêndios têm como atributos o número de bombeiros e o número de autotanques necessários para apagar o incêndio, que dependem do tipo de incêndio em questão. Os incêndios florestais necessitam apenas de ser socorridos por quartéis de bombeiros, visto que apenas necessitam de autotanques como veículos. Os veículos e socorristas necessários dependem da área que o incêndio toma: é necessário um bombeiro por 10 metros quadrados de área. Cada autotanque consegue levar 5 bombeiros. Os incêndios domésticos, para além de bombeiros, necessitam de agentes de polícia ou socorristas da INEM para passarem à organização dos espectadores do incêndio, de modo que ninguém pertencente ao público se magoe desnecessariamente, e atenderem às pessoas que tenham sofrido danos graças ao incêndio. Se o incêndio ocorrer num apartamento, serão necessários 4 bombeiros e 2 agentes de polícia / socorristas da INEM, e se o incêndio ocorrer numa moradia, precisa-se de 8 bombeiros para apagar o incêndio e 4 agentes de polícia / socorristas da INEM.

Os assaltos podem ocorrer em casas comerciais ou em casas particulares e destes podem resultar feridos. Para atender a este tipo de acidentes, precisa-se de agentes de polícia ou de INEM, dependendo de onde ocorreu o assalto: 4 agentes para casas comerciais e 2 para particulares. Se houver feridos, serão necessários tantos socorristas quantos feridos houver, tendo estes socorristas que vir em ambulâncias, para possibilitar levar os feridos para hospitais, caso seja necessário.

Os acidentes de viação podem ocorrer em auto-estradas ou em estradas nacionais. São necessários 4 agentes da polícia ou socorristas da INEM para atenderem aos feridos antes de chegarem ambulâncias, e organizarem a cena do acidente. São necessários dois socorristas que venham em ambulâncias para atender a cada veículo envolvido no acidente.

É possível adicionar um acidente novo à base de dados através da opção B do menu, e editar um acidente ainda não socorrido, em caso de engano ao submetê-lo ou de um aumento do número de feridos ou da área atingida por um incêndio desde o seu início, a partir da opção E.

2.3. Tratamento de Acidentes

Os acidentes de viação são tratados através da função `trataAcidentes()` que é chamada no início da aplicação, após serem abertos os ficheiros contendo a informação sobre acidentes e postos de socorro, sempre que é adicionado um novo acidente ou um novo posto à base de dados e quando é alterado um acidente que ainda não tenha sido socorrido ou um posto de socorro.

A função consiste em percorrer o vetor de acidentes do sistema, verificar que acidentes não foram tratados e, dependendo do tipo do acidente, chamar outra função: `verificaPostoMaisPerto()` que tem como argumentos o acidente e uma variável do tipo `bool` que representa se o posto é um posto de Polícia ou de Inem, e que utiliza ou carro ou mota. Se esta última variável for `false`, significa que queremos um posto de Bombeiros, com autotanques, para os incêndios, ou um posto de Bombeiros ou Inem cujo veículo seja a ambulância, para os assaltos e os acidentes de viação, para atender aos feridos. Caso a variável booleana seja `true`, significa que queremos um posto de Polícia ou de Inem, para atender à organização do acidente e ajudar no socorro das vítimas do acidente.

Dependendo então do acidente, a função vai verificar os recursos necessários para atender ao acidente, isto é, o número de socorristas e veículos, e percorrer o vetor de postos, à procura de um primeiro posto do tipo pretendido, que consiga corresponder aos recursos necessários. Caso esse primeiro posto não seja encontrado, não será possível socorrer o acidente, sendo lançada a exceção `SemRecursosDisponiveis`. Sendo encontrado, passa-se a verificar a existência de outros postos tal como este, comparando-os ao nível das distâncias, através da função `comparaDistancias()`, onde se utilizam os códigos postais.

Assim, encontra-se o posto mais próximo do acidente que tenha os recursos necessários para atender ao mesmo, pelo que se altera a variável booleana “tratado” do acidente para `true` e se coloca o acidente no vetor representativo do histórico de acidentes tratado pelos postos que socorreram este acidente.

2.4. Visualização de Informação

As classes `Acidente` e `PostoSocorro`, além dos métodos que retornam os atributos privados para efeitos de acesso a informação fora dessas classes, também possuem métodos virtuais que permitem a visualização por parte do utilizador dessa mesma informação.

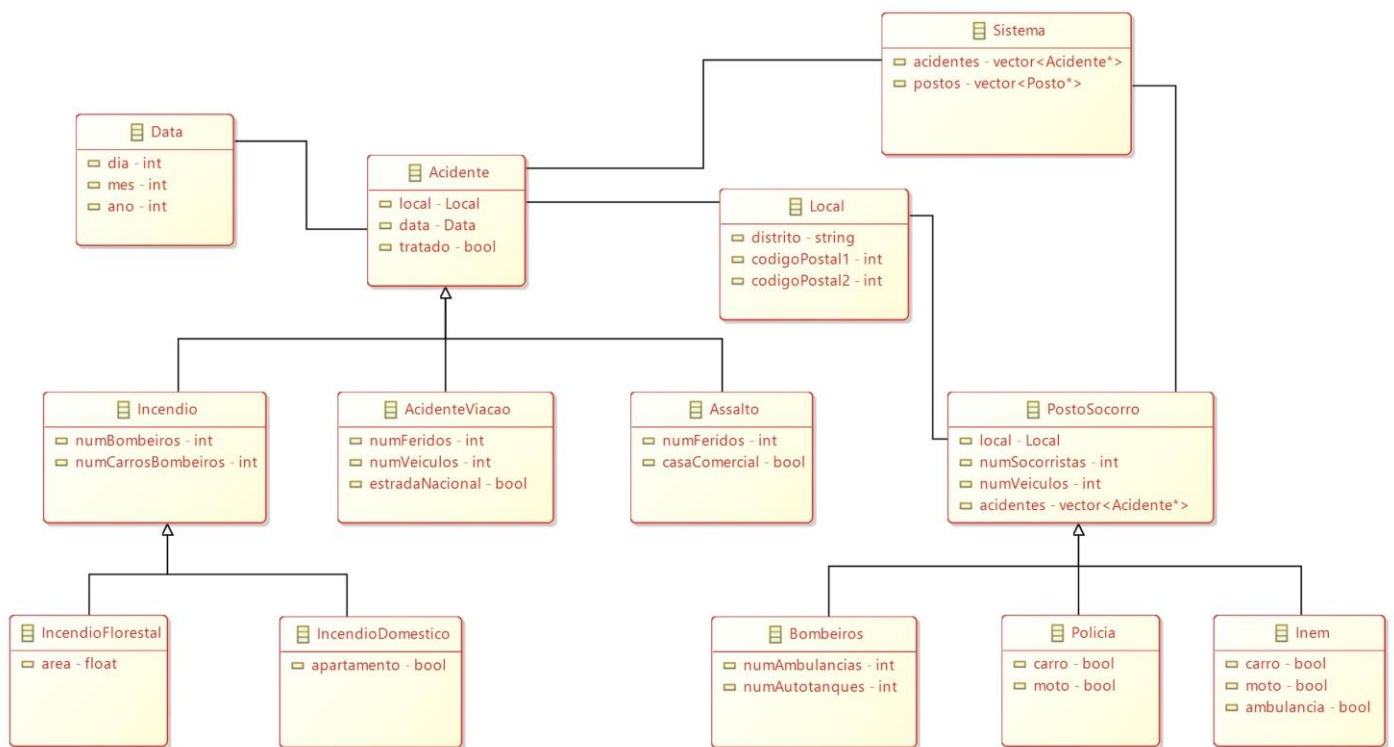
As funções de visualização são acessíveis através do menu inicial, estando disponíveis as seguintes:

- Visualização da informação dos postos de socorro existentes no sistema: para cada posto de socorro, pertencente ao vetor de postos do sistema, mostra-se toda a informação sobre a organização do posto.
- Visualização dos meios de socorro que atenderam aos acidentes: para cada acidente já socorrido, pertencente ao vetor de acidentes do sistema, são mostrados os postos que o socorreram, tal como os recursos disponibilizados para tal efeito.
- Visualização dos acidentes que ainda não foram socorridos: caso existam acidentes para os quais não existem recursos suficientes para os tratar, estes podem ser visualizados através desta função.
- Visualização de acidentes que ocorreram numa determinada data: mostra os acidentes que ocorreram na data especificada pelo utilizador e se foram socorridos ou não.
- Visualização dos acidentes que ocorreram num determinado distrito: mostra os acidentes que ocorreram no distrito especificado pelo utilizador.
- Visualização do histórico de acidentes de um posto: mostra todos os acidentes socorridos por um posto cujo tipo e código postal é especificado pelo utilizador.

2.5. Guardar Informação Alterada

Os postos de socorro criados pelo utilizador através da opção C ou alterados através da opção E do menu são adicionados ao ficheiro de texto “`postos.txt`” apenas no final do programa. No caso dos acidentes, são eliminados do ficheiro de texto “`acidentes.txt`” todos os acidentes já tratados, também no final do programa, sendo apenas guardados os acidentes que ainda não foram socorridos graças à falta de recursos.

3. Diagrama de Classes



4. Principais dificuldades

As principais dificuldades do projeto verificaram-se ao nível do tratamento dos acidentes por parte dos postos de socorro, tal como na identificação de alguns erros que ocorreram no decorrer do trabalho desenvolvido.

5. Contribuição de cada elemento do grupo

As funções realizadas no projeto foram divididas de forma equilibrada, de modo que ambos os membros do grupo participaram ativamente no desenvolvimento do mesmo. A classe correspondente aos acidentes foi desenvolvida pela aluna Helena Montenegro, enquanto que a classe correspondente aos postos de socorro ficou ao encargo da aluna Mariana Dias.

Deste modo, ambas as estudantes tiveram uma contribuição de 50% para o projeto realizado.