

T.D.A. Árvore Genealógica

Pretende-se construir um programa capaz de criar e gerir a árvore genealógica de uma dada família. O T.D.A. Árvore Genealógica (TDAAG) deve ter como elementos grafos, os nós são as pessoas, os arcos são as relações entre pessoas, a saber: a relação «progenitor» e a relação «cônjuge».

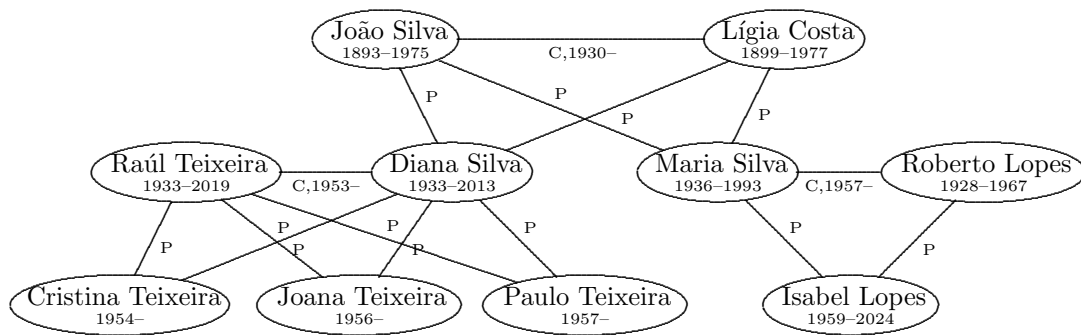


Figura 1: Árvore Genealógica (grafo) da Família Silva (fictícia)

Considere que a informação referente ao grafo é guardada num ficheiro, `familia.dados`,¹ a ser lido no início do programa. A informação contida no ficheiro servirá para criar o grafo como o representado acima (Ver Figura 1). O programa a construir deve ser capaz de ler (ou criar, caso não exista) um ficheiro desse tipo e no fim escrever a árvore genealógica de volta ao mesmo ficheiro.

Notas:

- As relações têm pesos (tipo, data início, data fim). Sendo os dois últimos opcionais;
- considera-se que pode haver divórcios;
- considera-se que não há duas relações diferentes entre o mesmo par de pessoas (não há casamentos entre progenitores e seus filhos);
- considera-se que não há lacetes;
- os dados pessoais a considerar são: nome; data de nascimento; data de falecimento.
- as funções referentes ao visualizar dos dados do grafo (`irmaos`, etc.) devem devolver como resultado uma lista de pessoas, usando para tal o T.D.A. `Lista` (de pessoas) em implementação simplesmente ligada.

Construa:

- O T.D.A. grafo, com a representação que achar mais conveniente, e com as operações: `criar`, `inserirPessoa`, `editarPessoa`, `removerPessoa`, `inserirRelacao`, `removerRelacao`. Além destas operações de gestão da estrutura de dados pretende-se que implemente as seguintes operações de pesquisa, para um dado ponto de partida do grafo: `irmaos`, `pais`, `ancestrais`, `familia`.

¹A estrutura do ficheiro é clara: a primeira linha contém o número de nós; depois segue-se a matriz de adjacências; finalmente tem-se a informação referente aos nós

- Um programa para fazer a gestão do grafo, isto é, utilizando o T.D.A. Grafo, um interface minimal, para ler/escrever do ficheiro e fazer a manipulação do grafo;
- Um programa para calcular a cadeia familiar mais longa. Utilize para tal o Algoritmo de Dijkstra [2].

Nota: não é necessário fazer nenhuma representação gráfica do grafo.

Datas

- Data de disponibilização do enunciado: 29/11/2024.
- Prazo para entrega: 10/1/2025 (às 24h00).

Material a Entregar

1. Documente o seu programa. Tanto em termos de documentação interna, como de documentação externa, esta última na forma de um pequeno manual de utilização (relatório, ≈ 10 pg.).
2. É obrigatório a organização do código em termos de um conjunto de ficheiros $\ll.h\gg$ e $\ll.c\gg$. É obrigatório a apresentação de uma `Makefile` que automatize o processo de compilação.
3. Deve entregar (por correio electrónico) um arquivo `zip`, contendo os ficheiros referentes ao programa (`Makefile`, `.c`, `.h`), assim como o ficheiro referente ao relatório (formato PDF), até às 24h00 do último dia do prazo. O nome do ficheiro a entregar deve ser `proj2MPIIGrpN.zip`, com N o número do grupo de trabalho.
4. Os código entregue pode ser desenvolvido em qualquer sistema computacional que contenha um compilador de C , mas deve compilar e os correspondentes programas devem funcionar nos computadores do laboratório de cálculo (sistemas Linux).

Referências

- [1] PEREIRA, JOSÉ MANUEL DOS SANTOS SIMÕES. *Grafos e redes: teoria e algoritmos básicos*. Rio de Janeiro : Interciência, 2013. (biblioteca DMUC - 94C/PER)
- [2] QUARESMA, PEDRO. *Métodos de Programação II*. Apontamentos da disciplina, Departamento de Matemática, Universidade de Coimbra, 2024.