

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA

Segundo Trabalho de Implementação

Alan Herculano Diniz
Rafael Belmock Perduzzi

Vitória
26 de novembro de 2018

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA

Alan Herculano Diniz
Rafael Belmock Perduzzi

Segundo Trabalho de Implementação

Relatório referente ao segundo trabalho de implementação da disciplina Programação III do curso de Ciência da Computação. UFES, Campus de Goiabeiras - Vitória.
Professor: João Paulo Almeida

Vitória
26 de novembro de 2018

Resumo

Este documento é um relatório do segundo trabalho de implementação da disciplina de Programação III do curso de Ciência da Computação da Universidade Federal do Espírito Santo, ministrada pelo professor João Paulo Almeida.

Sumário

	Sumário	3
	Introdução	4
1	DESENVOLVIMENTO	5
1.1	Leitor	5
1.2	Eleição	5
1.3	Coligação	5
1.4	Partido	5
1.5	Candidato	5
1.6	main	5
1.7	Bugs Conhecidos	5
1.8	CONSIDERAÇÕES FINAIS	6
	REFERÊNCIAS	7

Introdução

Neste segundo trabalho de implementação da disciplina de Programação III, foi necessário criar um leitor de arquivos de extensão CSV para retirar dados sobre eleições para vereadores. Para isso, foi necessário utilizar os conceitos de Programação Orientada a Objetos vistos em sala de aula. Para essa implementação, foi utilizada a linguagem de programação C++, que possui uma base forte de orientação a objetos.

1 DESENVOLVIMENTO

Neste capítulo, será descrita a implementação das classes utilizadas para o desenvolvimento do trabalho. Essas classes são as seguintes: Candidato, Partido, Coligação, Leitor e Eleição.

1.1 Leitor

O Leitor é responsável por abrir o arquivo CSV, escaneá-lo e interpretá-lo para então utilizar os dados lidos para instanciar objetos de outras classes. A operação de leitura retorna um objeto da classe Eleição.

1.2 Eleição

Eleição é a classe responsável por processar os dados lidos pelo Leitor para poder fazer a verificação de informações como: candidatos mais votados, quantidade de votos por partido e por coligação, candidatos eleitos, candidatos que não foram eleitos por causa do sistema proporcional e assim por diante.

1.3 Coligação

Um objeto da classe Coligação é aquele que possui e manipula informações sobre uma coleção de partidos, como quais partidos estão na Coligação e quantos votos essa Coligação possui. Também possui uma função de comparação para permitir a ordenação entre coligações.

1.4 Partido

Um objeto da classe Partido é análogo à Coligação, porém com uma coleção de candidatos em vez de partidos e um ponteiro para um objeto da classe coligação.

1.5 Candidato

A classe Candidato representa uma pessoa que se candidata a uma eleição, que está filiada a um partido e que possui uma certa quantidade de votos. Assim como em coligação e em partido, essa classe também possui uma função de comparação.

1.6 main

É a classe com o método de ponto de entrada do programa, que chama os métodos de outras classes e objetos para que o programa faça o que é necessário.

1.7 Bugs Conhecidos

Até o momento da entrega deste trabalho, não foi possível para os autores identificar qualquer problema ou bug no programa.

1.8 CONSIDERAÇÕES FINAIS

Com o desenvolvimento deste projeto, percebe-se a utilidade e a praticidade do uso do paradigma de orientação a objetos para o desenvolvimento de programas.

Referências

[1] Autor, Nome do, *Nome da obra*.