INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E

TECNOLOGIA DE SÃO PAULO

CAMPUS GUARULHOS

PROJETO FINAL – SISTEMA DE GERENCIAMENTO DE FUNCIONÁRIOS

RAFAEL PAIVA DIAS SILVA GU3005071

GUARULHOS

2020

**SUMÁRIO**

[1 INTRODUÇÃO 3](#_Toc52280199)

[2 SISTEMA 4](#_Toc52280200)

[3 BIBLIOTECAS UTILIZADAS NO PROJETO 5](#_Toc52280201)

[4 FUNÇÕES DESENVOLVIDAS 6](#_Toc52280202)

5 CONSIDERAÇÕES FINAIS……………………………………………………………… 9

# 1 INTRODUÇÃO

O Projeto Sistema de Gerenciamento de Funcionários foi pautado e desenvolvido de forma estruturada e modular com o embasamento teórico dos padrões de dados Lista Sequencial Dinâmica ou Lista Ligada (como, lista principal) e Lista Sequencial Estática (para lista de consultas).

No projeto foi considerado uma fábrica fictícia com uma lista de cargos padronizada pelo Código Brasileiro de Ocupações.

O Menu contempla as seguintes funcionalidades:

* Adicionar funcionário. O usuário que está utilizando o sistema efetua o registro de um funcionário, preenchendo os devidos campos, como: ID, nome, endereço, idade, salário e cargo. O último campo é formatado e já cadastrado segundo uma lista de cargos com o seu código e o seu cargo;
* Excluir funcionário. O usuário consegue excluir um registro de funcionário segundo o ID do funcionário. O sistema mostra os dados do funcionário do ID escolhido e pergunta ao usuário se deseja mesmo excluir o registro do funcionário. Após a confirmação do usuário, o sistema efetua a exclusão. Caso o ID não esteja cadastrado no sistema, o próprio informa que o ID é inexistente;
* Editar funcionário. Pelo ID escolhido, o usuário consegue verificar os campos de edição do funcionário. O usuário poderá editar os campos segundo a sua necessidade;
* Exibir funcionários ordenado por ID. Através do menu e a seleção de exibir funcionários por ID, o usuário poderá ver todos os funcionários ordenado pelos respectivos ID. Caso não haja funcionários cadastrados no sistema, o sistema irá informar;
* Exibir funcionários por faixa salarial. A partir desta funcionalidade, o usuário irá informar dois parâmetros, o limite inferior e superior de consulta do salário. Informando os dois limites, o sistema irá buscar os funcionários cadastrados dentre os parâmetros selecionados;
* Sair da aplicação. O Sistema finaliza a aplicação e fecha o programa.

# 2 SISTEMA

O sistema é estruturado em forma do TAD (Tipo Abstrato de Dados) e utiliza a Linguagem C em sua codificação.

O Sistema é composto:

* main.c. No main.c, só poderá chamar as funções do programa. Aqui comporta a função implementa\_cbo() que suscita uma lista referente aos códigos e cargos. O arquivo txt é compilado e transportado para a lista. Além disso, se tem no menu uma busca de tudo que será executado no programa;
* lista.h. Nesta parte estão o tipo de dados “Lista”. Foi criado uma struct que comporta os dados dos funcionários para acesso global que através do typedef sendo nomeada como FUNC e a estrutura elemento, como Lista;
* lista.c. As funções são específicas e modularizadas permanecendo a parte principal do projeto.;
* lista.txt. Este arquivo contempla o código dos funcionários e os seus respectivos cargos.

# 3 BIBLIOTECAS UTILIZADAS NO PROJETO

#include <stdio.h>

stdio.h é um cabeçalho da biblioteca padrão do C. Possui definições de subrotinas relativas às operações de entrada/saída, como leitura de dados digitados no teclado e exibição de informações na tela do programa de computador. Também possui numerosas definições de constantes, variáveis e tipos. É um dos cabeçalhos mais populares da linguagem de programação C, intensivamente utilizado tanto por programadores iniciantes como por experientes.

#include <stdlib.h>

stdlib.h é um arquivo cabeçalho da biblioteca de propósito geral padrão da linguagem de programação C. Ela possui funções envolvendo alocação de memória, controle de processos, conversões, entre outros.

#include <locale.h>

Tem o princípio de adaptar as características do programa a um determinado idioma ou região. Quando configurada corretamente garante que os caracteres especiais sejam exibidos normalmente pelo programa implementado.

#include <string.h>

A biblioteca string.h tem a finalidade de manipular strings e cadeias de caracteres e regiões de memória, além de fornecer funções e macros. Estas funções trabalham somente com cadeias de caracteres ASCII, contudo, não são compatíveis com Unicode.

#include "lista.h"

Biblioteca criada para arquivo TAD (Tipo Abstrato de Dados).

# 4 FUNÇÕES DESENVOLVIDAS

* int listaVazia(Lista \*li);

Verifica se a lista é Vazia.

* int insere\_lista\_ordenada(Lista \*li, FUNC al);

Insere os dados na lista de forma ordenada.

* Lista \*criaLista();

Cria a Lista.

* void implemanta();

Inicializa a lista.

* int remove\_lista(Lista \*li, int ID);

Exclui registro de dados da lista.

* int consultaID(Lista \*li, int ID, FUNC \*al);

Consulta na Lista através do ID.

* int consultaSalario(Lista \*li, int ID, float minimo,float maximo, FUNC \*al);

Consulta na lista pelos parâmetros informados pelo usuário.

* void apaga\_lista(Lista \*li);

Para destruir e liberar a memoria da lista.

* void titulo();

Cabeçalho do programa.

* void limpa\_buffer();

Limpa Qualquer Caractere que esteja na tela e funciona como system("pause").

* void menuPrincipal();

Função que chama a tela de menu.

* void escolha();

Função que verifica a escolha do usuário.

* void adicionar();

Função para adicionar registro.

* void excluir();

Função para excluir registro.

* void editar();

Funções para editar os registros.

* void exibir();

Funções para exibir os dados do registro do funcionário.

* void sair();

Função para sair do programa.

* void implementa\_cbo();

Função que corresponde a implementação de CBO.

* void inicializadorcbo(CBOLista \*cb);

Função que inicializa CBO.

* void separacao(char \*frase, CARGO \*funcionario);

Função que separa oque é codigo e oque é descrição do CBO.

* CBOLista \*cria\_cbo\_lista();

Função que cria a Lista para o CBO.

# 5 CONSIDERAÇÕES FINAIS

Foi realmente desafiador, talvez só com os slides e o material de apoio não conseguiria, porque foi essencial rever as aulas gravadas e fazer o passo-a-passo com as explicações dadas pelo professor Antônio Tartaglia, porém o maior desafio foi incluir um arquivo txt com os cargos CBO no programa, e fazer a lista estática conversar com o programa principal, tive q treinar apenas a lista estática várias vezes e acabar copiando o passo a passo dos slides mesmo, já a leitura com o FILE foi muito bom ver funcionando, confesso que tive que ver alguns documentos no site www.stackoverflow.com.b mas pra mim o importante e o grande alívio foi quando vi o programa de fato rodar e fazer todas as suas funções.