

Professor:	Ciro Cirne Trindade
Disciplina:	Linguagem de Programação
Curso:	Análise e Desenvolvimento de Sistemas
Grupos:	3 alunos
Data de entrega:	04/12/2019
Data da apresentação:	05/12/2019

2º Trabalho Prático – Controle de Vendas

Escreva um programa em C para controlar as vendas em uma loja de departamentos. O programa deve possuir as seguintes funcionalidades:

- Cadastro de clientes: deve permitir que sejam cadastradas as informações dos clientes da loja de departamentos: nome e telefone. Utilize a estrutura **cliente**, mostrada posteriormente, para representar as informações de um cliente. O código do cliente deve ser um número sequencial gerado automaticamente pelo sistema e mostrado no início do cadastro.
- Listagem dos clientes cadastrados: o sistema deve gerar uma listagem no formato de tabela de todos os clientes cadastrados, mostrando seu código, nome e telefone.
- Consulta por cliente: o sistema deve solicitar o nome de um cliente, e caso haja um ou mais clientes cadastrados com esse nome, deve mostrar o código, nome e o telefone desse(s) cliente(s). Caso não haja nenhum cliente com o nome fornecido, exibir uma mensagem. Não é necessário informar o nome completo do cliente, e o sistema deve exibir todos os clientes cujos nomes comecem com o valor fornecido. Por exemplo, se o usuário fornecer “Pedro” como nome de cliente procurado, o sistema de exibir os dados de todos os clientes cujos nomes comecem com “Pedro”.
- Cadastro de compras: deve permitir que sejam cadastradas as informações de uma compra realizada por um cliente previamente cadastrado. Utilize a estrutura **compra**, mostrada posteriormente, para representar as informações de uma compra. O sistema deve gerar o número da compra sequencialmente de forma automática e exibi-lo no início do cadastro. Depois deve solicitar o código do cliente e verificar se há um cliente cadastrado com este código. Caso negativo, exibir uma mensagem de erro e interromper a compra. Se houver um cliente cadastrado com o código fornecido, exibir o nome desse cliente e solicitar a data e o valor da compra. Não deve ser permitido informar um valor de compra menor ou igual a zero.
- Listagem das compras realizadas em uma data: o sistema deve solicitar uma data e mostrar o número da compra, o nome do cliente e o valor de todas as compras realizadas nesta data. No final, exibir o total vendido nesta data. O sistema deve possibilitar que esta listagem seja exibida no vídeo ou direcionada para um arquivo do tipo texto.
- Listagem das compras realizadas por um cliente: o sistema deve solicitar o código de um cliente e se houver um cliente cadastrado com este código, exibir o nome do cliente, a data e o valor de todas as compras realizadas por este cliente. No final, exibir o valor total comprado por este cliente. Se não houver um cliente cadastrado com o

código fornecido, exibir uma mensagem de erro. Se o cliente não realizou nenhuma compra, exibir uma mensagem.

O sistema deverá persistir os dados dos cadastros em arquivos. Forneça a implementação dos módulos definidos pelos arquivos *cliente.h* e *compra.h* mostrados a seguir. A função *main()* da aplicação deve ficar em um arquivo a parte.

```
/* cliente.h */

#ifndef _CLIENTE_H
#define _CLIENTE_H

/* Nome do arquivo aonde os clientes são cadastrados */
#define ARQ_CLIENTE "clientes.dat"

/* Estrutura que representa um cliente */
typedef struct {
    int codigo_cliente;
    char nome_cliente[51];
    char telefone[15];
} cliente;

/* função que cadastra um novo cliente no arquivo */
void cadastrar_cliente(void);

/* função que exibe uma listagem no formato de tabela
 * mostrando o código, nome e telefone de todos os
 * clientes cadastrados */
void listar_clientes(void);

/* função que consulta os clientes por nome */
void consultar_cliente(void);

#endif



---



/* compra.h */

#ifndef _COMPRA_H
#define _COMPRA_H

#include <stdbool.h>

/* Nome do arquivo aonde as compras são cadastradas */
#define ARQ_COMPRA "compras.dat"

/* Estrutura que representa uma data */
typedef struct {
    int dia;
    int mes;
    int ano;
} data;

/* Estrutura que representa uma compra */
typedef struct {
    int numero_compra;
    int codigo_cliente; // código do cliente que realizou a compra
    float valor;
    data dt_compra;
} compra;

/* função que cadastra uma nova compra no arquivo */
```

```
void cadastrar_compra(void);

/* função que lista todas as compras feitas em uma data */
void listar_compras_data(void);

/* função que devolve verdadeiro se as duas datas passadas
 * como argumentos são iguais, ou falso, caso contrário.
 * Usada na função listar_compras_data() */
bool datas_iguais(data, data);

/* função que lista todas as compras feitas por um cliente */
void listar_compras_cliente(void);

#endif
```

Informações importantes sobre Trabalho

1. Critérios de avaliação:
 - a) Código-fonte: 50%
 - i. Corretude: 70%
 - ii. Interface: 20%
 - iii. Legibilidade: 10%
 - b) Apresentação: 50%
2. Todos os trabalhos devem possuir no cabeçalho a identificação dos autores.
3. Trabalhos copiados (com ou sem eventuais disfarces) terão a nota dividida pelo número de cópias (inclusive o original).
4. Trabalhos atrasados não serão aceitos.
5. Trabalhos com erros de sintaxe (ou seja, erros de compilação) receberão nota ZERO.
6. É muito importante que seu programa tenha comentários e esteja bem indentado, ou seja, digitado de maneira a ressaltar a estrutura de subordinação dos comandos do programa. A legibilidade do código será levada em consideração pelo critério de avaliação do trabalho.
7. Você deve enviar os fontes (.c e .h) e o *makefile* da aplicação para o seguinte e-mail: ciroct@gmail.com
8. Guarde uma cópia do e-mail enviado pelo menos até o final do semestre. Este material pode ser solicitado pelo professor em caso de extravio.