

ALUNO: PIETRO GOUDEL FAVORETO

RA: 2021102557

Lista de Exercícios Struct

1. Crie um programa em C que preencha uma struct cliente usando scanf depois do preenchimento imprima-a:

```
//#####
```

```
//# Autor: Pietro
```

```
//# Exercício 1
```

```
//#####
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
struct CLIENTE {
```

```
char nome [20];
```

```
int idade;
```

```
};
```

```
int main(void) {
```

```
struct CLIENTE cliente;
```

```
printf("Ensira o seu nome\n");
```

```
scanf("%s", &cliente.nome);
```

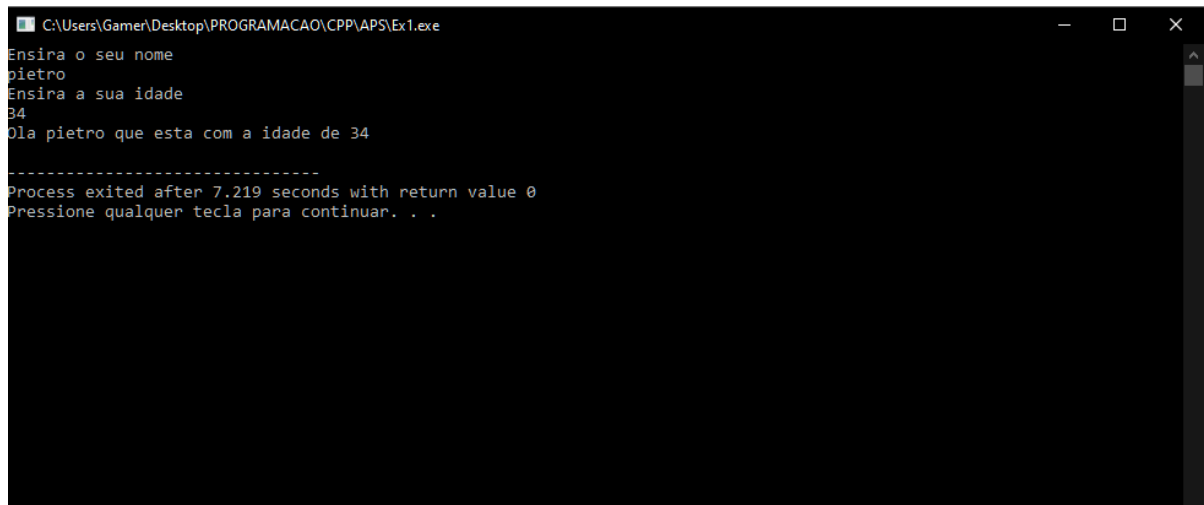
```
system("CLS");
```

```
printf("Ensira a sua idade\n");
```

```
scanf("%i", &cliente.idade);
```

```
system("CLS");
```

```
printf("Ola %s que esta com a idade de %i\n", cliente.nome, cliente.idade);  
}
```



```
C:\Users\Gamer\Desktop\PROGRAMACAO\CPP\APS\Ex1.exe  
Ensira o seu nome  
pietro  
Ensira a sua idade  
34  
Ola pietro que esta com a idade de 34  
  
-----  
Process exited after 7.219 seconds with return value 0  
Pressione qualquer tecla para continuar. . .
```

2. Crie um programa em C que preencha uma struct CADASTRO com duas substructs ENDERECO (comercial e residencial) usando scanf, depois do preenchimento imprima a matriz e todo seu conteúdo:

```
//#####
```

```
//# Autor: Pietro
```

```
//# Exercicio 2
```

```
//#####
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
struct ENDC {
```

```
char cidade [50];
```

```
char bairro [20];
```

```
char rua [50];
```

```
int numero;
```

```
};
```

```
struct ENDR {
```

```
char cidade [50];
```

```
char bairro [20];
```

```
char rua [50];
```

```
int numero;
```

```
};
```

```
struct CADASTRO {
```

```
char nome [20];
```

```
int idade;
```

```
struct ENDC enderecocomercial;
```

```
struct ENDR enderecoresidencial;
```

```
};
```

```
int main(void) {
```

```
struct CADASTRO cliente;
```

```
//Cadastro
```

```
printf("Ensira o seu nome\n");
```

```
scanf("%s", &cliente.nome);
```

```
printf("\n");
```

```
printf("Ensira a sua idade\n");
```

```
scanf("%i", &cliente.idade);
```

```
system("CLS");
```

```
//ENDComercial
```

```
printf("Ensira sua cidade comercial\n");
```

```
scanf("%s", &cliente.enderecocomercial.cidade);
```

```
printf("\n");
```

```
printf("Ensira o seu bairro comercial\n");
```

```
scanf("%s", &cliente.enderecocomercial.bairro);
```

```
printf("\n");
```

```
printf("Ensira a sua rua comercial\n");
```

```
scanf("%s", &cliente.enderecocomercial.rua);
```

```
printf("\n");
```

```
printf("Ensira o seu numero comercial\n");
```

```
scanf("%i", &cliente.enderecocomercial.numero);
```

```
system("CLS");
```

```
//ENDResidencial

printf("Ensira sua cidade residencial\n");
scanf("%s", &cliente.enderecoresidencial.cidade);
printf("\n");

printf("Ensira o seu bairro residencial\n");
scanf("%s", &cliente.enderecoresidencial.bairro);
printf("\n");

printf("Ensira a sua rua residencial\n");
scanf("%s", &cliente.enderecoresidencial.rua);
printf("\n");

printf("Ensira o seu numero residencial\n");
scanf("%i", &cliente.enderecoresidencial.numero);
system("CLS");

printf("Ola %s que esta com %i anos\n\n", cliente.nome, cliente.idade);

printf("Seu cadastro foi completado\n\nComercial:\n\nCidade: %s\nBairro: %s\nRua: %s\nNumero: %i", cliente.enderecocomercial.cidade, cliente.enderecocomercial.bairro, cliente.enderecocomercial.rua, cliente.enderecocomercial.numero);

printf("\n\nResidencial:\n\nCidade: %s\nBairro: %s\nRua: %s\nNumero: %i", cliente.enderecoresidencial.cidade, cliente.enderecoresidencial.bairro, cliente.enderecoresidencial.rua, cliente.enderecoresidencial.numero);
}
```

```
C:\Users\Gamer\Desktop\PROGRAMACAO\CPP\APS\Ex2.exe
Insira o seu nome
peitro

Insira a sua idade
47
Insira sua cidade comercial
cidadeeee

Insira o seu bairro comercial
bairro

Insira a sua rua comercial
ruaaa

Insira o seu numero comercial
66
Insira sua cidade residencial
cidadezinha

Insira o seu bairro residencial
bairroroaj

Insira a sua rua residencial
ruaiia

Insira o seu numero residencial
98
Ola peitro que esta com 47 anos

Seu cadastro foi completado

Comercial:
Cidade: cidadeeee
Bairro: bairro
Rua: ruaaa
Numero: 66

Residencial:
Cidade: cidadezinha
Bairro: bairroroaj
Rua: ruaiia
Numero: 98
-----
Process exited after 31.21 seconds with return value 0
Pressione qualquer tecla para continuar. . .
```

3. Crie um vetor com 5 structs CADASTRO com duas substructs ENDERECO (comercial e residencial), preencha cada um com scanf e depois imprima. Importante que existam dois loops distintos, um para o preenchimento e outro para a impressão.

```
//#####
```

```
//# Autor: Pietro
```

```
//# Exercicio 3
```

```
//#####
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
struct ENDC {
```

```
char cidade [50];
```

```
char bairro [20];
```

```
char rua [50];
```

```
int numero;
```

```
};
```

```
struct ENDR {
```

```
char cidade [50];
```

```
char bairro [20];
```

```
char rua [50];
```

```
int numero;
```

```
};
```

```
struct CADASTRO {
```

```
char nome [20];
```

```
int idade;
```

```
struct ENDC enderecomercial;
```

```
struct ENDR enderecoresidencial;
```

```
};
```

```
int main(void) {  
    struct CADASTRO cliente[5];  
  
    for(int x = 0;x < 5;x++){  
        //Cadastro  
        printf("Ensira o seu nome\n");  
        scanf("%s", &cliente[x].nome);  
        printf("\n");  
  
        printf("Ensira a sua idade\n");  
        scanf("%i", &cliente[x].idade);  
        system("CLS");  
  
        //ENDComercial  
        printf("Ensira sua cidade comercial\n");  
        scanf("%s", &cliente[x].enderecocomercial.cidade);  
        printf("\n");  
  
        printf("Ensira o seu bairro comercial\n");  
        scanf("%s", &cliente[x].enderecocomercial.bairro);  
        printf("\n");  
  
        printf("Ensira a sua rua comercial\n");  
        scanf("%s", &cliente[x].enderecocomercial.rua);  
        printf("\n");  
  
        printf("Ensira o seu numero comercial\n");  
        scanf("%i", &cliente[x].enderecocomercial.numero);  
        system("CLS");  
  
        //ENDResidencial
```



```
printf("Ensira sua cidade residencial\n");
scanf("%s", &cliente[x].enderecoresidencial.cidade);
printf("\n");
```

```
printf("Ensira o seu bairro residencial\n");
scanf("%s", &cliente[x].enderecoresidencial.bairro);
printf("\n");
```

```
printf("Ensira a sua rua residencial\n");
scanf("%s", &cliente[x].enderecoresidencial.rua);
printf("\n");
```

```
printf("Ensira o seu numero residencial\n");
scanf("%i", &cliente[x].enderecoresidencial.numero);
system("CLS");
}
```

```
for(int x = 0; x < 5; x++){
```

```
printf("Ola %s que esta com %i anos\n\n", cliente[x].nome, cliente[x].idade);

printf("Seu cadastro foi completado\n\nComercial:\n\nCidade: %s\nBairro: %s\nRua: %s\nNumero: %i", cliente[x].enderecocomercial.cidade, cliente[x].enderecocomercial.bairro, cliente[x].enderecocomercial.rua, cliente[x].enderecocomercial.numero);

printf("\n\nResidencial:\n\nCidade: %s\nBairro: %s\nRua: %s\nNumero: %i", cliente[x].enderecoresidencial.cidade, cliente[x].enderecoresidencial.bairro, cliente[x].enderecoresidencial.rua, cliente[x].enderecoresidencial.numero);

}

}
```

Exemplo do print, mudado a variável para 2, para ficar mais fácil e cabível

```
C:\Users\Gamer\Desktop\PROGRAMACAO\CPP\APS\Ex3.exe
Ensira o seu bairro comercial
bairoro

Ensira a sua rua comercial
ruooooi

Ensira o seu numero comercial
82
Ensira sua cidade residencial
cidoda

Ensira o seu bairro residencial
baioro

Ensira a sua rua residencial
ruaaa

Ensira o seu numero residencial
77
Ola pietro que esta com 67 anos

Seu cadastro foi completado

Comercial:

Cidade: ciadeal
Bairro: bairrooooo
Rua: ruaaaaoao
Numero: 78

Residencial:

Cidade: cidadao
Bairro: bairooorr
Rua: ruaiiooaa
Numero: 900la joao que esta com 23 anos

Seu cadastro foi completado

Comercial:

Cidade: cidaade
Bairro: bairoro
Rua: ruooooi
Numero: 82

Residencial:

Cidade: cidoda
Bairro: baioro
Rua: ruaaa
Numero: 77
-----
Process exited after 66.14 seconds with return value 0
Pressione qualquer tecla para continuar. . .
```

4. Crie duas structs CORRESPONDENCIA (com CEP, rua, numero, bairro, cidade, estado, uma será chamada de A e outra de B. Preencha os dados da A, copie os dados para a struct B e imprima a struct B.

```
//#####
```

```
//# Autor: Pietro
```

```
//# Exercicio 4
```

```
//#####
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
struct CORRESPONDENCIA {
```

```
int cep;
```

```
char rua [50];
```

```
int numero;
```

```
char bairro [50];
```

```
char cidade [50];
```

```
char estado [50];
```

```
};
```

```
int main(void) {
```

```
struct CORRESPONDENCIA A, B;
```

```
printf("Ensira o cep da correspondencia\n");
```

```
scanf("%i", &A.cep);
```

```
system("CLS");
```

```
printf("Ensira o estado da correspondencia\n");
```

```
scanf("%s", &A.estado);
```

```
system("CLS");
```

```
printf("Ensira a cidade da correspondencia\n");
```

```
scanf("%s", &A.cidade);
```

```
system("CLS");
```

```
printf("Ensira o bairro da correspondencia\n");
```

```
scanf("%s", &A.bairro);
```

```
system("CLS");
```

```
printf("Ensira a rua da correspondencia\n");
```

```
scanf("%s", &A.rua);
```

```
system("CLS");
```

```
printf("Ensira o numero da correspondencia\n");
```

```
scanf("%i", &A.numero);
```

```
system("CLS");
```

```
B=A;
```

```
printf("Correspondencia B\n\nCEP: %i\nEstado: %s\nCidade: %s\nBairro: %s\nRua: %s\nNumero: %i", B.cep, B.estado, B.cidade, B.bairro, B.rua, B.numero);
```

```
}
```

```
C:\Users\Gamer\Desktop\PROGRAMACAO\CPP\APS\Ex4.exe
Ensira o cep da correspondencia
124325
Ensira o estado da correspondencia
estadonovo
Ensira a cidade da correspondencia
cidadeadopovo
Ensira o bairro da correspondencia
bairrobaixo
Ensira a rua da correspondencia
ruaparaguai
Ensira o numero da correspondencia
8756
Correspondencia B

CEP: 124325
Estado: estadonovo
Cidade: cidadeadopovo
Bairro: bairrobaixo
Rua: ruaparaguai
Numero: 8756
-----
Process exited after 25.59 seconds with return value 0
Pressione qualquer tecla para continuar. . .
```

5. Crie um vetor com 5 structs CADASTRO (com nome e idade), cadastre os dados em um loop, em um segundo loop busque a struct que tem a maior idade e imprima.

```
//#####

//# Autor: Pietro

//# Exercicio 5

//#####
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
struct CADASTRO {
```

```
char nome[50];
```

```
int idade;
```

```
};
```

```
int main(void) {
```

```
struct CADASTRO cliente[5];
```

```
int maiorIdade = 0;
```

```

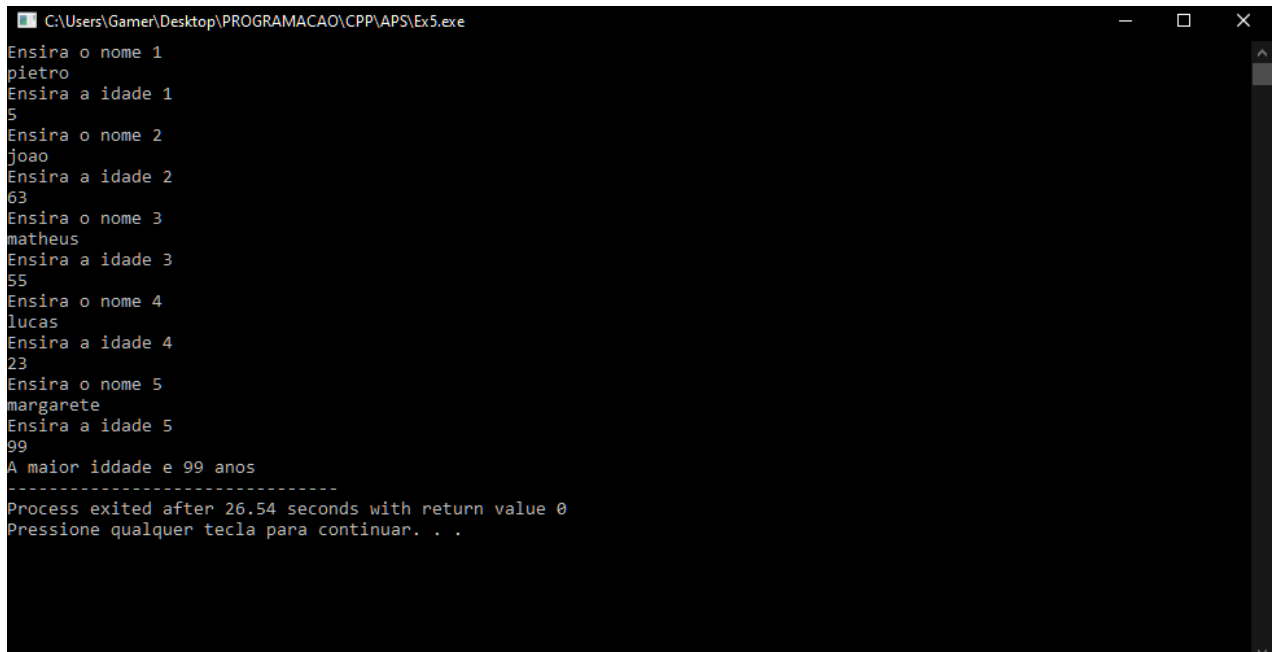
for(int x = 0;x < 5;x++){
printf("Ensira o nome %i\n", (x + 1));
scanf("%s", &cliente[x].nome);
system("CLS");

printf("Ensira a idade %i\n", (x + 1));
scanf("%i", &cliente[x].idade);
system("CLS");
}

for(int x = 0;x < 5;x++){
if(cliente[x].idade > maiorIdade){
maiorIdade = cliente[x].idade;
}
}

printf("A maior idade e %i anos", maiorIdade);
}

```



```

C:\Users\Gamer\Desktop\PROGRAMACAO\CPP\APS\Ex5.exe
Ensira o nome 1
pietro
Ensira a idade 1
5
Ensira o nome 2
joao
Ensira a idade 2
63
Ensira o nome 3
matheus
Ensira a idade 3
55
Ensira o nome 4
lucas
Ensira a idade 4
23
Ensira o nome 5
margarete
Ensira a idade 5
99
A maior iddade e 99 anos
-----
Process exited after 26.54 seconds with return value 0
Pressione qualquer tecla para continuar. . .

```

6. Faça um programa em C que preencha um vetor de struct com a seguinte estrutura:

Nome

Cargo

Telefone

e-mail

Salário

O programa devera:

- a) Cadastrar 5 funcionários, um após o outro.
- b) Imprimir o total dos salários de todos os funcionários
- c) Informar qual o funcionário que possui o maior salário
- d) Informar se algum funcionário está com o numero de telefone em branco

```
//#####
```

```
//# Autor: Pietro
```

```
//# Exercicio 6
```

```
//#####
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
struct FUNC {  
char nome[50];  
char cargo[20];  
int telefone;  
char email[50];  
int salario;  
};
```

```
int main(void) {  
struct FUNC funcionario[5];  
int somaSalarios = 0;  
int maiorSalario = 0;  
int indiceMaiorSalario;
```

```
for(int x = 0; x < 5; x++){  
printf("Ensira o nome do funcionario:%i\n", (x + 1));  
scanf("%s", &funcionario[x].nome);  
system("CLS");
```

```
printf("Ensira o cargo do funcionario: %i\n", (x + 1));  
scanf("%s", &funcionario[x].cargo);  
system("CLS");
```

```
printf("Ensira o telefone do funcionario: %i\n", (x + 1));  
scanf("%i", &funcionario[x].telefone);  
system("CLS");
```

```
printf("Ensira o email do funcionario: %i\n", (x + 1));  
scanf("%s", &funcionario[x].email);  
system("CLS");
```



```
printf("Ensira o salario do funcionario: %i\n", (x + 1));  
scanf("%i", &funcionario[x].salario);  
system("CLS");  
}
```

```
for(int x = 0;x < 5;x++){  
somaSalarios = somaSalarios + funcionario[x].salario;  
}
```

```
printf("Soma dos salarios: %i\n", somaSalarios);
```

```
for(int x = 0;x < 5;x++){  
if(funcionario[x].salario > maiorSalario){  
maiorSalario = funcionario[x].salario;  
indiceMaiorSalario = x;  
}  
}
```

```
printf("Maior salario: %i, do funcionario: %s\n", funcionario[indiceMaiorSalario].salario,  
funcionario[indiceMaiorSalario].nome);
```

```
for(int x = 0;x < 5;x++){  
if(funcionario[x].telefone == '-1')  
printf("Funcionario %s, esta com telefone vazio", funcionario[x].nome);  
}  
  
}
```

```
C:\Users\Gamer\Desktop\PROGRAMACAO\CPP\APS\Ex6.exe
Insira o nome do funcionario:1
pietro
Insira o cargo do funcionario: 1
gerente
Insira o telefone do funcionario: 1
827618323
Insira o email do funcionario: 1
pietro@email
Insira o salario do funcionario: 1
4500
Insira o nome do funcionario:2
joao
Insira o cargo do funcionario: 2
operador
Insira o telefone do funcionario: 2
46768233
Insira o email do funcionario: 2
joao@email
Insira o salario do funcionario: 2
2000
Soma dos salarios: 6500
Maior salario: 4500, do funcionario: pietro

-----
Process exited after 29.87 seconds with return value 0
Pressione qualquer tecla para continuar. . .
```

7. Crie uma agenda telefônica como no exemplo 6, um sistema de menus deverá cadastrar cada usuário. (1. Novo 2. Remover, 3. Buscar, 4. Editar, 5. Sair)

```
//#####
```

```
//# Autor: Pietro
```

```
//# Exercício 7
```

```
//#####
```

```
// ANOTACOES:
```

```
// Professor, tentei fazer desta maneira de criar varias funcoes para cada uma das opcoes
```

```
// mas acabei falhando e pedi ajuda pro chat gpt kkkk, pois nao foi algo ainda passado em aula
```

```
// entao analisei o codigo dele e fiz da maneira que eu precisava.
```

```
// Achei interessante o uso do #define, pois ao em vez de mudarmos no projeto inteiro, por exemplo
```

// um maximo de contatos, apenas muda no inicio da variavel ali, e pronto, muda em tudo que utilizava.

// Percebi que para utilizar uma funcao, a sua criacao deve estar acima da linha em que é chamada

// no caso as funções são criadas antes, e depois são chamada no int main(), la em baixo.

// Em C#, que eu utilizo no trabalho, acaba podendo criar a função em qualquer parte do código, e depois chamar ela em qualquer parte também

// Como a busca pelo nome do usuario seria mais para essa criação, fiz por ela, mas criei uma função a mais, que mostra todos os contatos registrados

// fgets(); lê uma linha inteira de entrada do usuário, que o senhor nao ensinou ainda

// getchar(); limpa o buffer do \n deixado pelo scanf

// system("CLS"); limpar console

// não sei se o nome disso é função mesmo, mas acabo falando assim no trabalho kkkk

#include <stdio.h>

#include <stdlib.h> // utilizar o CLS para limpar o console

#include <string.h> // utlizaar fgets e getchar

// Caso queira deixar o maximo de contatos, alterar por aqui, fica mais tranquilo

#define MAX_CONTATOS 5

#define MAX_NOME 50

#define MAX_TELEFONE 20

struct CONTATO {

 char nome[MAX_NOME];

 char telefone[MAX_TELEFONE];

```
};
```

```
// função para aparecer as opções do menu, em vez de deixar isso setado no MAIN()
```

```
void exibir_menu() {  
    printf("\n");  
    printf("1. Adicionar contato\n");  
    printf("2. Buscar contato\n");  
    printf("3. Editar contato\n");  
    printf("4. Deletar contato\n");  
    printf("5. Mostrar todos os contatos\n");  
    printf("6. Sair\n");  
    printf("\nEscolha uma opcao: ");  
}
```

```
// adiciona um ccontato caso N seja menos que MAX_CONTATOS
```

```
int adicionar_contato(struct CONTATO lista[], int n) {  
    if (n >= MAX_CONTATOS) {  
        system("CLS");  
        printf("Lista de contatos cheia.\n");  
        return n;  
    }  
    printf("Digite o nome: ");  
    fgets(lista[n].nome, MAX_NOME, stdin);  
    printf("Digite o telefone: ");  
    fgets(lista[n].telefone, MAX_TELEFONE, stdin);  
    n++;  
    system("CLS");  
    printf("Contato adicionado com sucesso.\n");  
    return n;  
}
```

```
// Busca o contato trazendo seu nome e telefone, buscnando pelo seu nome
```

```
void buscar_contato(struct CONTATO lista[], int n) {  
    char nome[MAX_NOME];  
    printf("Digite o nome a buscar: ");  
    fgets(nome, MAX_NOME, stdin);  
    for (int i = 0; i < n; i++) {  
        if (strcmp(lista[i].nome, nome) == 0) {  
            system("CLS");  
            printf("Contato encontrado:\n");  
            printf("Nome: %s", lista[i].nome);  
            printf("Telefone: %s", lista[i].telefone);  
            return;  
        }  
    }  
    system("CLS");  
    printf("Contato nao encontrado.\n");  
}
```

```
// Busca todos os contatos trazendo seu nome e telefone
```

```
void buscar_todos(struct CONTATO lista[], int n) {  
    if(n > 0){  
        for (int i = 0; i < n; i++) {  
            printf("Contato %i encontrado:\n", (i + 1));  
            printf("Nome: %s", lista[i].nome);  
            printf("Telefone: %s\n\n", lista[i].telefone);  
        }  
        return;  
    }  
    system("CLS");  
    printf("Nenhum contato encontrado.\n");  
}
```

```
// Edita um contato buscando ele pelo seu nome
int editar_contato(struct CONTATO lista[], int n) {
    char nome[MAX_NOME];
    printf("Digite o nome a editar: ");
    fgets(nome, MAX_NOME, stdin);
    for (int i = 0; i < n; i++) {
        if (strcmp(lista[i].nome, nome) == 0) {
            printf("Digite o novo nome: ");
            fgets(lista[i].nome, MAX_NOME, stdin);
            printf("Digite o novo telefone: ");
            fgets(lista[i].telefone, MAX_TELEFONE, stdin);
            system("CLS");
            printf("Contato editado com sucesso.\n");
            return 1;
        }
    }
    system("CLS");
    printf("Contato nao encontrado.\n");
    return 0;
}
```

```
// Deleta o contato buscando ele por seu nome
int deletar_contato(struct CONTATO lista[], int n) {
    char nome[MAX_NOME];
    printf("Digite o nome a deletar: ");
    fgets(nome, MAX_NOME, stdin);
    for (int i = 0; i < n; i++) {
        if (strcmp(lista[i].nome, nome) == 0) {
            for (int j = i; j < n - 1; j++) {
                strcpy(lista[j].nome, lista[j+1].nome);
            }
        }
    }
}
```

```

        strcpy(lista[j].telefone, lista[j+1].telefone);
    }
    n--;
    system("CLS");
    printf("Contato deletado com sucesso.\n");
    return n;
}
}
system("CLS");
printf("Contato nao encontrado.\n");
return n;
}

```

// rotina rodada em loop sempre que inicia, até que o usuário escolha o numero 5 nas opções

```

int main() {
    CONTATO lista[MAX_CONTATOS];
    int quantidadeContatos = 0;
    int opcao;

    do {
        exibir_menu();
        scanf("%d", &opcao);
        getchar();
    } while (opcao != 5);
    system("CLS");
    switch (opcao) {
        case 1:
            quantidadeContatos = adicionar_contato(lista, quantidadeContatos);
            break;
        case 2:
            buscar_contato(lista, quantidadeContatos);
            break;
    }
}

```

```
case 3:
    editar_contato(lista, quantidadeContatos);
    break;
case 4:
    quantidadeContatos = deletar_contato(lista, quantidadeContatos);
    break;
case 5:
    buscar_todos(lista, quantidadeContatos);
    break;
case 6:
    system("CLS");
    printf("Saindo...\n");
    break;
default:
    system("CLS");
    printf("Opcao invalida.\n");
    break;
}
} while (opcao != 6);

return 0;
}
```


C:\Users\Gamer\Desktop\PROGRAMACAO\CPP\APS\Ex7.exe

```
1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair
```

Escolha uma opcao: 5
Nenhum contato encontrado.

```
1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair
```

Escolha uma opcao: 4
Digite o nome a deletar: pietro
Contato nao encontrado.

```
1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair
```

Escolha uma opcao: 3
Digite o nome a editar: pietro
Contato nao encontrado.

```
1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair
```

Escolha uma opcao: 2
Digite o nome a buscar: pietro
Contato nao encontrado.

```
1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair
```

Escolha uma opcao:

C:\Users\Gamer\Desktop\PROGRAMACAO\CPP\APS\Ex7.exe

1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 1
Digite o nome: pietro
Digite o telefone: 4672673
Contato adicionado com sucesso.

1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 2
Digite o nome a buscar: pietro
Contato encontrado:
Nome: pietro
Telefone: 4672673

1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 3
Digite o nome a editar: pietro
Digite o novo nome: joao
Digite o novo telefone: 26637373
Contato editado com sucesso.

1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 2
Digite o nome a buscar: joao
Contato encontrado:
Nome: joao
Telefone: 26637373

1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 5
Contato 1 encontrado:

```
C:\Users\Gamer\Desktop\PROGRAMACAO\CPP\APS\Ex7.exe
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 1
Digite o nome: pietro
Digite o telefone: 376362
Contato adicionado com sucesso.

1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 4
Digite o nome a deletar: pietro
Contato deletado com sucesso.

1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 5
Nenhum contato encontrado.

1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 2
Digite o nome a buscar: pietro
Contato nao encontrado.

1. Adicionar contato
2. Buscar contato
3. Editar contato
4. Deletar contato
5. Mostrar todos os contatos
6. Sair

Escolha uma opcao: 6
Saindo...

-----
Process exited after 22.1 seconds with return value 0
Pressione qualquer tecla para continuar. . .
```