



**CENTRO UNIVERSITÁRIO INTERNACIONAL UNINTER
ESCOLA SUPERIOR POLITÉCNICA
CST ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
LINGUAGEM DE PROGRAMAÇÃO**

ATIVIDADE PRÁTICA

**ANA LÍCIA SANTOS BRAGA – RU: 3423922
PROFESSOR WINSTON SEM LUN FUNG**

**MONTES CLAROS – MG
2021**

1 PRÁTICA 1

Escreva um algoritmo em linguagem C que atenda os seguintes requisitos:

- ✓ Crie um registro para armazenar o seu Nome de um funcionário e o seu salário.
- ✓ Solicite ao usuário que digite o Nome Completo e o salário.
- ✓ Crie um ponteiro para o registro.
- ✓ Através do ponteiro para o registro verifique a alíquota do imposto de renda para o salário informado, utilize a tabela abaixo para determinar a alíquota:

| Base de cálculo | Alíquota |
|--------------------------|----------|
| de 0,00 até 1.903,98 | isento |
| de 1.903,99 até 2.826,65 | 7,50% |
| de 2.826,66 até 3.751,05 | 15,00% |
| de 3.751,06 até 4.664,68 | 22,50% |
| a partir de 4.664,68 | 27,50% |

- ✓ Através do ponteiro para o registro imprima o Nome Completo, o salário, a alíquota do imposto de renda (em %) e o valor a ser deduzido do salário.

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

struct info //cria a struct para guardar os dados
{
    char nome[50];
    float salario;
}; struct info dados, *reg; //cria a variável do tipo da struct criada e o ponteiro para registro

int main()
{
    setlocale(LC_ALL, "Portuguese"); //configura para receber caracteres do teclado ABNT2
    char c;
    reg = &dados;

    printf("Digite o nome do funcionário:\n");
    fgets(reg->nome, 50, stdin); //recebe e guarda o nome do funcionario
    while ((c = getchar()) != '\n' && c != EOF) {} //limpa o buffer de teclado

    printf("Digite o salário do funcionário em R$:\n");
    scanf_s("%f", &reg->salario); //recebe e guarda o salário do funcionário
    while ((c = getchar()) != '\n' && c != EOF) {} //limpa o buffer de teclado

    if (reg->salario > 0) //verifica se o salario informado é válido
    {
        if (reg->salario <= 1903.98) //estabelece as condições de cada faixa de alíquota e imprime os dados solicitados
        {
            printf("Dados do funcionário:\n");
            printf("Nome completo: %s\n", reg->nome);
            printf("Salário recebido: R$%.2f\n", reg->salario);
            printf("Alíquota devida: Isento.\n");
        }
        else if (reg->salario >= 1903.99 && reg->salario < 2826.65)
        {
            printf("Dados do funcionário:\n");
            printf("Nome completo: %s\n", reg->nome);
            printf("Salário recebido: R$%.2f\n", reg->salario);
        }
    }
}
```

```

        printf("Alíquota devida: 7,5%\n");
        printf("Valor a ser deduzido: R$%.2f\n\n", reg->salario * 0.075);
    }
    else if (reg->salario >= 2826.66 && reg->salario < 3751.05)
    {
        printf("Dados do funcionário:\n");
        printf("Nome completo: %s\n", reg->nome);
        printf("Salário recebido: R$%.2f\n", reg->salario);
        printf("Alíquota devida: 15%\n");
        printf("Valor a ser deduzido: R$%.2f\n\n", reg->salario * 0.15);
    }
    else if (reg->salario >= 2751.06 && reg->salario < 4664.68)
    {
        printf("Dados do funcionário:\n");
        printf("Nome completo: %s\n", reg->nome);
        printf("Salário recebido: R$%.2f\n", reg->salario);
        printf("Alíquota devida:22,5%\n");
        printf("Valor a ser deduzido: R$%.2f\n\n", reg->salario * 0.225);
    }
    else
    {
        printf("Dados do funcionário:\n");
        printf("Nome completo: %s\n", reg->nome);
        printf("Salário recebido: R$%.2f\n", reg->salario);
        printf("Alíquota devida:27,5%\n");
        printf("Valor a ser deduzido: R$%.2f\n\n", reg->salario * 0.275);
    }
}
else
{
    printf("Valor de salário inválido!\n\n");
}

system("pause");
return 0;
}

```

Para demonstrar o funcionamento faça as capturas de tela do terminal utilizando seu nome completo e o no campo salário utilize se RU conforme indicado abaixo:

| | 1 | 2 | 3 | 4 | 5 | , | 6 | 7 |
|-------------|--------------------|---|---|---|---|---|---|------------------|
| Salário R\$ | Primeiro dígito | | | | | , | | Último dígito |

```

Microsoft Visual Studio
Console de Depuração do Microsoft Visual Studio

Digite o nome do funcionário:
Ana Licia Santos Braga

Digite o salário do funcionário em R$:
34239,22
Dados do funcionário:
Nome completo: Ana Licia Santos Braga

Salário recebido: R$34239,22
Alíquota devida:27,5%
Valor a ser deduzido: R$9415,79

Pressione qualquer tecla para continuar. . .

O C:\Users\Ana Licia\Documents\CURSOS\UNINTER\Curso Análise
e Des sistemas\D12 - Linguagem de Progrmação\Projetos\Ativid
adePratica\Debug\AtividadePratica.exe (processo 16324) foi e
ncerrado com o código 0.
Para fechar o console automaticamente quando a depuração par
ar, habilite Ferramentas -> Opções -> Depuração -> Fechar o
console automaticamente quando a depuração parar.
Pressione qualquer tecla para fechar esta janela...

```

2 PRÁTICA 2

Escreva um algoritmo em linguagem C que atenda os seguintes requisitos:

- ✓ Crie um vetor com a quantidade de dígitos do seu RU.
- ✓ Solicite que usuário digite o seu RU, cada dígito digitado deve ser armazenado em uma posição do vetor. Por exemplo:

| | | | | | | | |
|----------|-----------------|---|---|---|---|---|---------------|
| Vetor RU | Primeiro dígito | | | | | | Último dígito |
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |

- ✓ Utilizando ponteiros verifique o valor da soma de todos os elementos contidos neste vetor e imprima na tela.

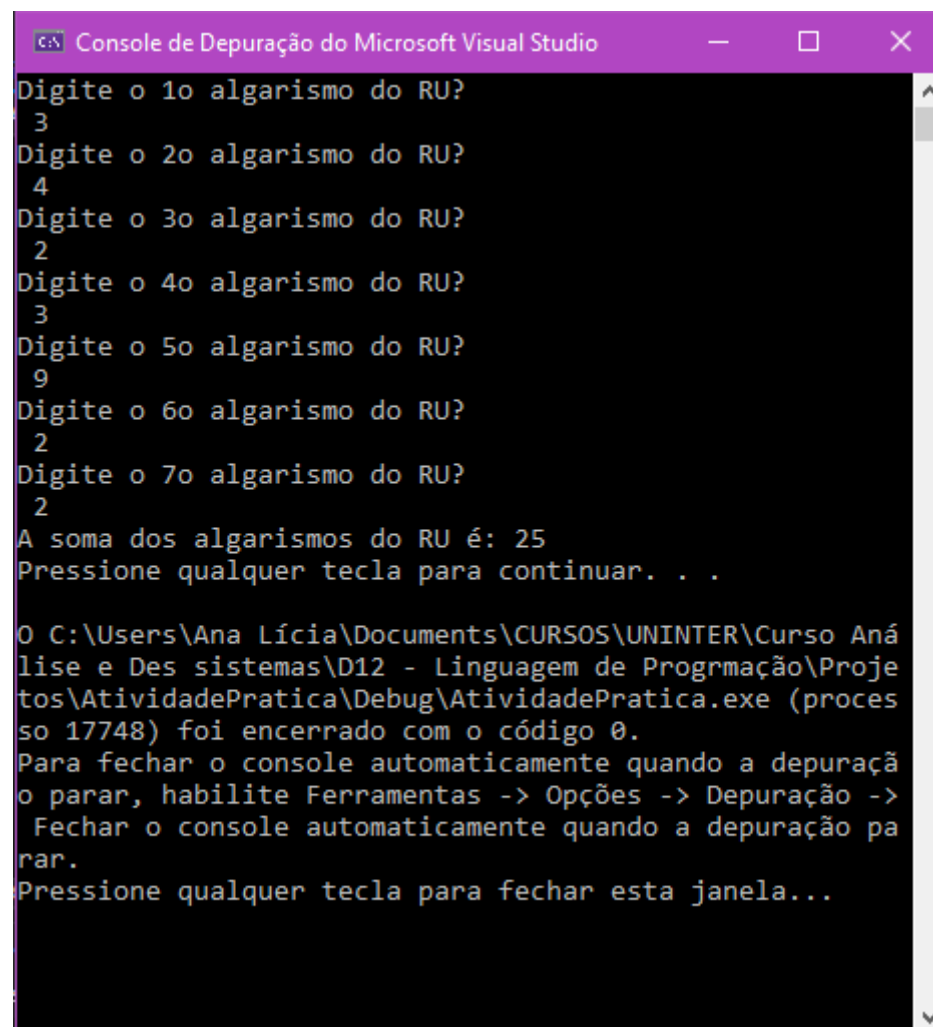
```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

int main()
{
    setlocale(LC_ALL, "Portuguese"); //configura para receber caracteres do teclado ABNT2
    int RU[7], i, soma = 0;
    char c;
    int* pon;

    for (i = 0; i < 7; i++)
    {
        printf("Digite o %do algarismo do RU?\n ", i + 1); //recebe os algarismos
        scanf_s("%d", &RU[i]); //guarda o algarismo no vetor
        pon = &RU[i]; //transmite o valor do vetor para o ponteiro
        soma = soma + (*pon); //faz a soma utilizando o ponteiro
        while ((c = getchar()) != '\n' && c != EOF) {} //limpa o buffer de teclado
    }
    printf("A soma dos algarismos do RU é: %d\n", soma);

    system("pause");
    return 0;
}
```

Para demonstrar o funcionamento faça as capturas de tela do terminal utilizando seu RU.



```
CS Console de Depuração do Microsoft Visual Studio
Digite o 1o algarismo do RU?
3
Digite o 2o algarismo do RU?
4
Digite o 3o algarismo do RU?
2
Digite o 4o algarismo do RU?
3
Digite o 5o algarismo do RU?
9
Digite o 6o algarismo do RU?
2
Digite o 7o algarismo do RU?
2
A soma dos algarismos do RU é: 25
Pressione qualquer tecla para continuar. . .

O C:\Users\Ana Lícia\Documents\CURSOS\UNINTER\Curso Análise e Des sistemas\D12 - Linguagem de Progrmação\Projetos\AtividadePratica\Debug\AtividadePratica.exe (processo 17748) foi encerrado com o código 0.
Para fechar o console automaticamente quando a depuração parar, habilite Ferramentas -> Opções -> Depuração -> Fechar o console automaticamente quando a depuração parar.
Pressione qualquer tecla para fechar esta janela...
```


3 PRÁTICA 3

Faça um programa, em linguagem C, para calcular a soma de duas matrizes 4 x 4 de números inteiros. O programa deve implementar uma função chamada **calc_soma** que calcula a soma de duas matrizes.

O programa deve solicitar ao usuário que seja informado as matrizes **A** e **B** de dimensões 4 x4, calcular e imprimir a **matriz resultado C** da soma das matrizes **A** com **B**. O programa termina quando for digitado um valor negativo. A função deve obedecer ao seguinte protótipo:

```
void calc_soma(int *mat_A, int *mat_B, int *mat_C);
```

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

int A[4][4]; //declara as variáveis das matrizes 4x4
int B[4][4];
int C[4][4];

void calc_soma(int* mat_A, int* mat_B, int* mat_C); //declara o protótipo da função calc_soma

int main()
{
    setlocale(LC_ALL, "Portuguese"); //configura para receber caracteres do teclado ABNT2
    printf("    Soma de matrizes\n");
    printf("    Ana Lícia Braga - RU 3423922\n");
    int i, j;
    char ch;
    printf("\n    Defina a Matriz A:\n");
    for (i = 0; i < 4; i++){
        for (j = 0; j < 4; j++){
            printf("Digite o valor %d|%d = ", i+1, j+1); //preenche a Matriz A
```

```

scanf_s("%d", &A[i][j]);
while ((ch = getchar()) != '\n' && ch != EOF) {} //limpa o buffer de teclado
if (A[i][j] < 0)
{
    printf("Somente números positivos! Programa encerrado!"); //encerra o programa caso número negativo
    return 1;
}
}
}
printf("\n    Defina a Matriz B:\n");
for (i = 0; i < 4; i++) {
    for (j = 0; j < 4; j++) {
        printf("Digite o valor %d|%d = ", i+1, j+1);
        scanf_s("%d", &B[i][j]);
        while ((ch = getchar()) != '\n' && ch != EOF) {} //limpa o buffer de teclado
        if (B[i][j] < 0)
        {
            printf("Somente números positivos! Programa encerrado!"); //encerra o programa caso número negativo
            return 1;
        }
    }
}

calc_soma(*A, *B, *C); //chama a função para cálculo da soma de matrizes
printf("\n    Matriz A\n"); //imprime a matriz A
for (int i = 0; i < 4; i++) {
    for (int j = 0; j < 4; j++) {
        printf("%i\t", A[i][j]);
    }
    printf("\n");
}
printf("\n    Matriz B\n"); //imprime a matriz B
for (i = 0; i < 4; i++) {
    for (j = 0; j < 4; j++) {
        printf("%d\t", B[i][j]);
    }
    printf("\n");
}
printf("\n    Matriz C = Matriz A + Matriz B\n"); //imprime a matriz C (resultado da soma de A e B)
for (i = 0; i < 4; i++){

```

```
        for (j = 0; j < 4; j++){
            printf("%d\t", C[i][j]);
        }
        printf("\n");
    }
    system("pause");
    return 0;
}

void calc_soma(int* mat_A, int* mat_B, int* mat_C) // declara a função calc_soma
{
    int i;
    for (i = 0; i < 16; i++) { //percorre a matriz (que quando passada por ponteiro, torna-se um vetor unidimensional)
        mat_C[i] = mat_A[i] + mat_B[i]; //faz a soma da matrizes
    }

    return; //retorna para a função principal sem enviar valores (void)
}
```

Para demonstrar o funcionamento faça os print de tela.

```

C:\Users\Ana Lícia\Documents\CURSOS\UNINTER\Curso Aná...
Soma de matrizes
Ana Lícia Braga - RU 3423922

Defina a Matriz A:
Digite o valor 1|1 = 1
Digite o valor 1|2 = 2
Digite o valor 1|3 = 3
Digite o valor 1|4 = 4
Digite o valor 2|1 = 5
Digite o valor 2|2 = 6
Digite o valor 2|3 = 7
Digite o valor 2|4 = 8
Digite o valor 3|1 = 9
Digite o valor 3|2 = 1
Digite o valor 3|3 = 2
Digite o valor 3|4 = 3
Digite o valor 4|1 = 4
Digite o valor 4|2 = 56
Digite o valor 4|3 = 7
Digite o valor 4|4 = 8

Defina a Matriz B:
Digite o valor 1|1 = 9
Digite o valor 1|2 = 1
Digite o valor 1|3 = 2
Digite o valor 1|4 = 3
Digite o valor 2|1 = 4
Digite o valor 2|2 = 5
Digite o valor 2|3 = 6
Digite o valor 2|4 = 7
Digite o valor 3|1 = 8
Digite o valor 3|2 = 9
Digite o valor 3|3 = 1
Digite o valor 3|4 = 2
Digite o valor 4|1 = 3
Digite o valor 4|2 = 4
Digite o valor 4|3 = 5
Digite o valor 4|4 = 6

Matriz A
1      2      3      4
5      6      7      8
9      1      2      3

```

```

Console de Depuração do Microsoft Visual Studio
Digite o valor 4|1 = 3
Digite o valor 4|2 = 4
Digite o valor 4|3 = 5
Digite o valor 4|4 = 6

Matriz A
1      2      3      4
5      6      7      8
9      1      2      3
4      56     7      8

Matriz B
9      1      2      3
4      5      6      7
8      9      1      2
3      4      5      6

Matriz C = Matriz A + Matriz B
10     3      5      7
9      11     13     15
17     10     3      5
7      60     12     14

Pressione qualquer tecla para continuar. . .

O C:\Users\Ana Lícia\Documents\CURSOS\UNINTER\Curso Análise
e Des sistemas\D12 - Linguagem de Progrmação\Projetos\Ativid
adePratica\Debug\AtividadePratica.exe (processo 17728) foi e
ncerrado com o código 0.
Para fechar o console automaticamente quando a depuração par
ar, habilite Ferramentas -> Opções -> Depuração -> Fechar o
console automaticamente quando a depuração parar.
Pressione qualquer tecla para fechar esta janela...

```

4 PRÁTICA 4

Faça, em linguagem C, uma função recursiva para o cálculo multiplicação. Sabe-se que o cálculo pode ser feito através de somas sucessivas.

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

int* v; // define o ponteiro do vetor v

int preencherVetor(int* x, int y, int z); //declara a função recursiva

int main()
{
    setlocale(LC_ALL, "Portuguese"); //configura para receber caracteres do teclado ABNT2
    int i, n, m=0;
    char c;
    printf("    Calculadora de soma recursiva\n");
    printf("    Ana Lícia Braga - RU 3423922\n");

    //define o tamanho do vetor
    printf("Digite o fator multiplicador:\n"); //solicita ao usuário o tamanho do vetor(quantidade de posições)
    scanf_s("%d", &n);
    while ((c = getchar()) != '\n' && c != EOF) {} //limpa o buffer de teclado

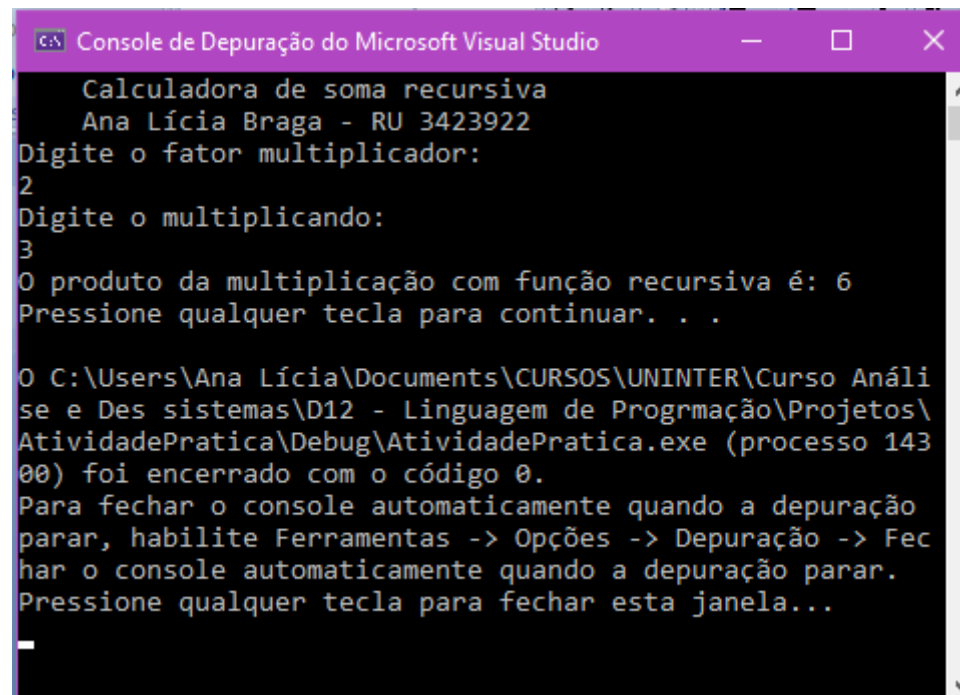
    //preenche o vetor
    printf("Digite o multiplicando:\n"); //solicita ao usuário os valores das posições do vetor
    scanf_s("%d", &m);
    while ((c = getchar()) != '\n' && c != EOF) {} //limpa o buffer de teclado

    //apresenta o resultado da função que preenche o vetor com os dados coletados, passados por parâmetro
    printf("O produto da multiplicação com função recursiva é: %i\n", preencherVetor(v, n, m));

    system("pause");
    return 0;
}
```

```
int preencherVetor(int *x, int y, int z){ //função recursiva recebe os valores preenchidos pelo usuário na main
    int i, soma=0;
    x = (int *) malloc(y * sizeof(int)); //estabelece o tamanho do vetor com a alocação de memória necessária
    for (i=0; i<y; i++) { //preenche as posições do vetor e faz a soma recursivamente
        x[i] = z;
        soma += x[i];
    }
    return soma;
}
```

Para demonstrar o funcionamento utilize o primeiro dígito do seu RU como multiplicando e o último dígito como multiplicador.



```
Microsoft Visual Studio
Console de Depuração do Microsoft Visual Studio

Calculadora de soma recursiva
Ana Lícia Braga - RU 3423922
Digite o fator multiplicador:
2
Digite o multiplicando:
3
O produto da multiplicação com função recursiva é: 6
Pressione qualquer tecla para continuar. . .

O C:\Users\Ana Lícia\Documents\CURSOS\UNINTER\Curso Análise e Des sistemas\D12 - Linguagem de Progrmação\Projetos\AtividadePratica\Debug\AtividadePratica.exe (processo 14300) foi encerrado com o código 0.
Para fechar o console automaticamente quando a depuração parar, habilite Ferramentas -> Opções -> Depuração -> Fechar o console automaticamente quando a depuração parar.
Pressione qualquer tecla para fechar esta janela...
_
```

5 PRÁTICA 5

Crie um programa, em linguagem C, que receba 6 registros contendo, Nome, CPF, telefone e email. Solicite que sejam digitados todos os dados de todos os registros e ao final salve-os em um arquivo.csv, utilize o ; (**ponto e vírgula**) para separador de campo. O nome do arquivo deve ser o seu número de RU.

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>
#include <string.h>

struct info { //declaração da struct contendo os dados solicitados
    char nome[41];
    char CPF[13]; //informado long int na aula ao vivo, mas não comporta o tamanho de um CPF. Usei char para simplificar a tarefa
    char telefone[12];
    char email[41];
}; struct info dados[6]; //variável do tipo da struct declarada como vetor para 6 posições

FILE* p_arq; //ponteiro para arquivo

int main()
{
    setlocale(LC_ALL, "Portuguese"); //configura para receber caracteres do teclado ABNT2

    printf("    Manipulação de arquivo CSV\n");
    printf("Ana Lícia Braga - RU 3423922\n");
    int x;
    x = fopen_s(&p_arq, "RU3423922.csv", "w"); //abre o arquivo para escrita

    if (x == NULL) //verifica se o arquivo foi aberto no comando anterior
    {
        int i;
        for (i = 0; i < 6; i++) { //preenche as posições do vetor e faz a soma recursivamente
            printf("Digite o nome do %io contato:\n", i+1);
            fgets(dados[i].nome, 41, stdin); //recebe e guarda o nome do contato
            dados[i].nome[strcspn(dados[i].nome, "\n")] = 0;
        }
    }
}
```

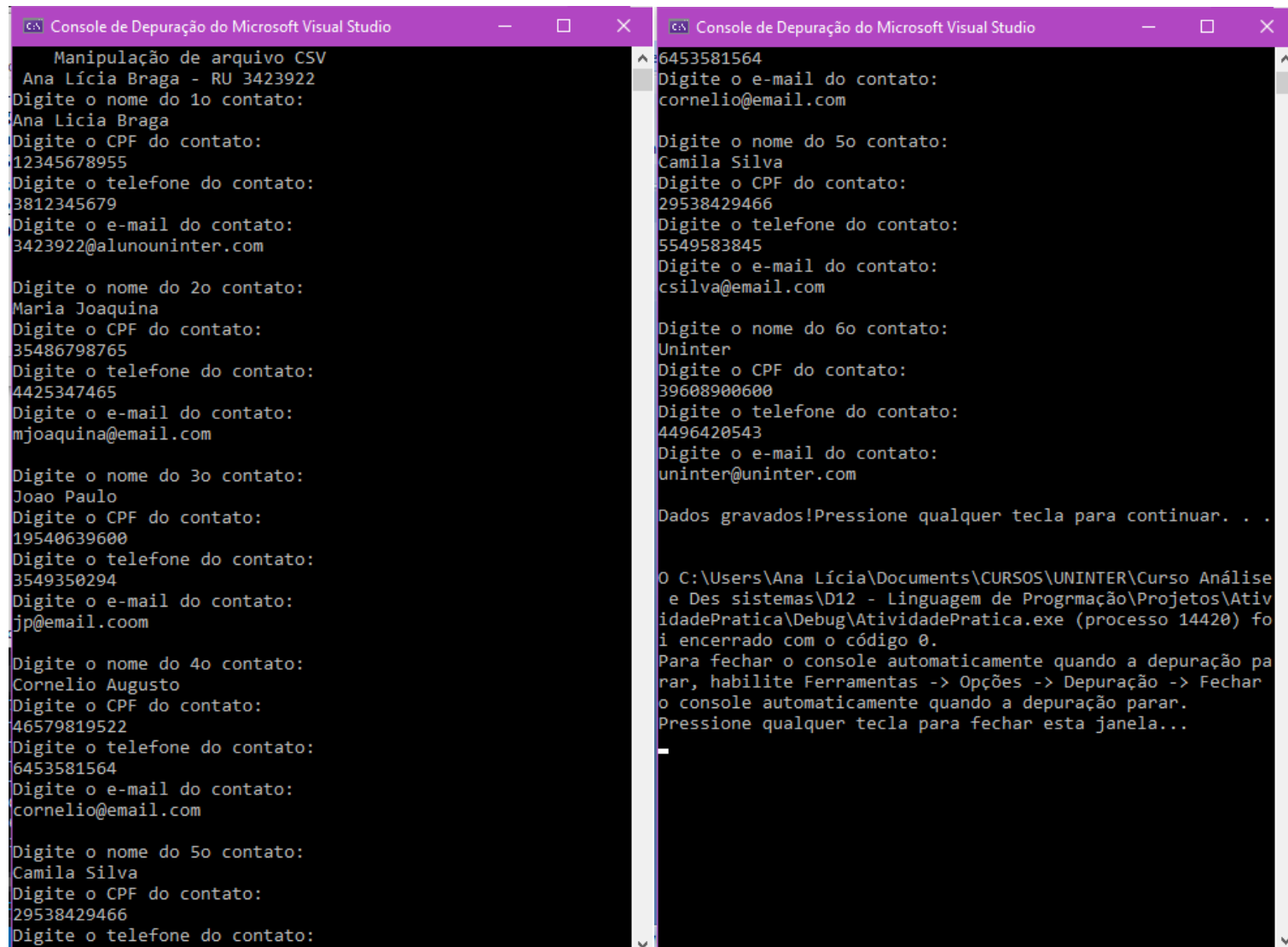
```

printf("Digite o CPF do contato:\n");
fgets(dados[i].CPF, 13, stdin); //recebe e guarda o CPF do contato
dados[i].CPF[strcspn(dados[i].CPF, "\n")] = 0;

printf("Digite o telefone do contato:\n");
fgets(dados[i].telefone, 12, stdin); //recebe e guarda o telefone do contato
dados[i].telefone[strcspn(dados[i].telefone, "\n")] = 0;

printf("Digite o e-mail do contato:\n");
fgets(dados[i].email, 41, stdin); //recebe e guarda o e-mail do contato
dados[i].email[strcspn(dados[i].email, "\n")] = 0;
printf("\n");
}
fprintf(p_arq, "Nome; CPF; Telefone; E-mail;\n"); //grava os dados no arquivo com o separador ;
for (i = 0; i < 6; i++)
{
    fprintf(p_arq, "%s;%s;%s;%s\n", dados[i].nome, dados[i].CPF, dados[i].telefone, dados[i].email);
}
fclose(p_arq); //fecha o arquivo com os dados gravados
printf("Dados gravados!");
}
else //caso o arquivo retorne algum problema para abrir
{
    printf("Erro na abertura do arquivo!\n");
    return 1;
}
system("pause");
return 0;
}

```

```
Console de Depuração do Microsoft Visual Studio
Manipulação de arquivo CSV
Ana Lícia Braga - RU 3423922
Digite o nome do 1o contato:
Ana Lícia Braga
Digite o CPF do contato:
12345678955
Digite o telefone do contato:
3812345679
Digite o e-mail do contato:
3423922@alunouninter.com

Digite o nome do 2o contato:
Maria Joaquina
Digite o CPF do contato:
35486798765
Digite o telefone do contato:
4425347465
Digite o e-mail do contato:
mjoaquina@email.com

Digite o nome do 3o contato:
Joao Paulo
Digite o CPF do contato:
19540639600
Digite o telefone do contato:
3549350294
Digite o e-mail do contato:
jp@email.com

Digite o nome do 4o contato:
Cornelio Augusto
Digite o CPF do contato:
46579819522
Digite o telefone do contato:
6453581564
Digite o e-mail do contato:
cornelio@email.com

Digite o nome do 5o contato:
Camila Silva
Digite o CPF do contato:
29538429466
Digite o telefone do contato:
6453581564
Digite o e-mail do contato:
cornelio@email.com

Digite o nome do 5o contato:
Camila Silva
Digite o CPF do contato:
29538429466
Digite o telefone do contato:
29538429466
Digite o e-mail do contato:
csilva@email.com

Digite o nome do 6o contato:
Uninter
Digite o CPF do contato:
39608900600
Digite o telefone do contato:
4496420543
Digite o e-mail do contato:
uninter@uninter.com

Dados gravados!Pressione qualquer tecla para continuar. . .

O C:\Users\Ana Lícia\Documents\CURSOS\UNINTER\Curso Análise
e Des sistemas\D12 - Linguagem de Progrmação\Projetos\Ativ
idadePratica\Debug\AtividadePratica.exe (processo 14420) fo
i encerrado com o código 0.
Para fechar o console automaticamente quando a depuração pa
rar, habilite Ferramentas -> Opções -> Depuração -> Fechar
o console automaticamente quando a depuração parar.
Pressione qualquer tecla para fechar esta janela...
```

| AtividadePratica | | | Pesquisar AtividadePratica |
|----------------------------------|---------------------|-----------------------|----------------------------|
| Nome | Data de modificação | Tipo | |
| .vs | 13/05/2021 09:16 | Pasta de arquivos | |
| Debug | 17/05/2021 18:26 | Pasta de arquivos | |
| ++ AtividadePratica.cpp | 17/05/2021 15:08 | C++ Source | |
| AtividadePratica.sln | 13/05/2021 09:16 | Visual Studio Solu... | |
| AtividadePratica.vcxproj | 13/05/2021 11:32 | VC++ Project | |
| AtividadePratica.vcxproj.filters | 13/05/2021 11:32 | VC++ Project Filte... | |
| AtividadePratica.vcxproj.user | 13/05/2021 09:16 | Per-User Project O... | |
| ++ Origem.cpp | 17/05/2021 18:26 | C++ Source | |
| RU3423922.csv | 17/05/2021 18:29 | Planilha OpenOffi... | |

RU3423922.csv - Bloco de Notas

```

Arquivo Editar Formatar Exibir Ajuda
Nome; CPF; Telefone; E-mail;
Ana Licia Braga;12345678955;3812345679;3423922@alunouninter.com
Maria Joaquina;35486798765;4425347465;mjoaquina@email.com
Joao Paulo;19540639600;3549350294;jp@email.com
Cornelio Augusto;46579819522;6453581564;cornelio@email.com
Camila Silva;29538429466;5549583845;csilva@email.com
Uninter;39608900600;4496420543;uninter@uninter.com

```

Ln 1, Col 1 100% Windows (CRLF) UTF-8