

CENTRO UNIVERSITÁRIO INTERNACIONAL UNINTER

ESCOLA SUPERIOR POLITÉCNICA

BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO

DISCIPLINA DE LINGUAGEM DE PROGRAMAÇÃO

ATIVIDADE PRÁTICA

ARNALDO ROCHA FILHO – RU: 3112775

prof. WINSTON sen lun FUNG

SÃO MATEUS DO SUL – PARANÁ

2020

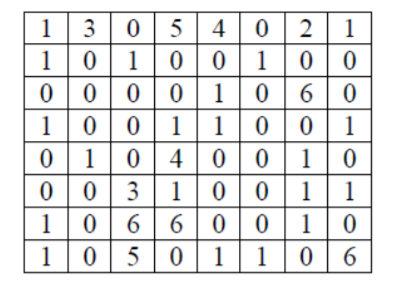
# EXERCÍCIO

ENUNCIADO: Vamos supor que várias pedras do jogo de xadrez estão no tabuleiro. Para facilitar a indicação das peças, vamos convencionar:

**1** – peões  **3** – torres  **5** – reis **0** – ausência de peças

**2** – cavalos **4** – bispos **6** – rainhas

O tabuleiro é o seguinte:



Construa um programa, em linguagem C, que:

a) Solicite ao usuário que informe qual peça deseja colocar em cada posição do tabuleiro.

b) Mostre o tabuleiro informado pelo usuário na tela.

c) Determine a qual a quantidade de cada tipo de peça no tabuleiro

Solução do aluno:

**Não esqueça de explicar o código com comentários!!**

**CTRL+C/CTRL+V do Visual Studio!!**

**Se desejar, defina as páginas do documento para PAISAGEM para melhor inserir os códigos.**

#include <stdio.h>

#include <stdlib.h>

#include <locale.h>

enum pecas { vazio, peao, cavalo, torre, bispo, rei, rainha }; // enum para usar como contador

int main()

{

setlocale(LC\_ALL, "Portuguese"); // com isso, ele faz funcionar todas as funções da lingua portuguesa, assim ele aceita todas acentuações.

printf("Olá! Eu sou Arnaldo Rocha Filho, aluno da uninter. Então vamos brincar com o tabulheiro?! \n\n");

system("pause");

system("cls"); // limpa a janela para que não acumule, ficando assim mais bonito

int tabuleiro[8][8] = { // variável para formar a matriz.

{1, 3, 0, 5, 4, 0, 2, 1},

{1, 0, 1, 0, 0, 1, 0, 0},

{0, 0, 0, 0, 1, 0, 6, 0},

{1, 0, 0, 1, 1, 0, 0, 1},

{0, 1, 0, 4, 0, 0, 1, 0},

{0, 0, 3, 1, 0, 0, 1, 1},

{1, 0, 6, 6, 0, 0, 1, 0},

{1, 0, 5, 0, 1, 1, 0, 6}

};

printf("O tabulheiro se inicia desta forma!\n\n"); // este printf serve para imprimir na tela instruções para o usuário seguir

int L, C; //variavél para fazer laço de repetição para ser possivel a impressão da matriz na tela

for (L = 0; L < 8; L++)

{

for (C = 0; C < 8; C++)

{

printf("%d ", tabuleiro[L][C]); //imprimir na tela o tabulheiro

}

printf("\n");

}

int cont[7] = { 0 }; // variável com vetor contador com 7 e que pode inserir um valor

int a,b; //variavel para forma a matriz

for (a = 0; a < 8; a++) //laço de repetição para inserir valor em cada caso

{

for (b = 0; b < 8; b++) {

switch (tabuleiro[a][b])

{

case vazio:

cont[0]++;

break;

case peao:

cont[1]++;

break;

case cavalo:

cont[2]++;

break;

case torre:

cont[3]++;

break;

case bispo:

cont[4]++;

break;

case rei:

cont[5]++;

break;

case rainha:

cont[6]++;

break;

default:

break;

}

}

}

printf("\n");

printf("Cada número abaixo representa uma peça.\n\n"); // este printf serve para imprimir na tela instruções para o usuário seguir

printf("1 = Peão\n2 = Cavalo\n3 = Torre\n4 = Bispo\n5 = Rei\n6 = Rainha \n\n");

printf("Aonde tem o ZERO, significa que ali é ausente de peça!\n\n");// este printf serve para imprimir na tela instruções para o usuário seguir

system("pause");

system("cls");

printf("Quantidade de peças:\n\n");

printf("Total de Peoes = %d\n", cont[1]);

printf("Total de Cavalos = %d\n", cont[2]);

printf("Total de Torres = %d\n", cont[3]);

printf("Total de Bispos = %d\n", cont[4]);

printf("Total de Reis = %d\n", cont[5]);

printf("Total de Rainhas = %d\n", cont[6]);

//imprimi o total de peças na matriz individualmente

printf("Total geral das peças é = %d\n", cont[1]+cont[2]+cont[3]+cont[4]+cont[5]+cont[6]); //soma e imprimi o total de peças

printf("\n");

system("pause");

cont[0] = 0; // zera todos os contadores

cont[1] = 0;

cont[2] = 0;

cont[3] = 0;

cont[4] = 0;

cont[5] = 0;

cont[6] = 0;

int tabuleiroEscolhido[8][8] = {

{1, 3, 0, 5, 4, 0, 2, 1},

{1, 0, 1, 0, 0, 1, 0, 0},

{0, 0, 0, 0, 1, 0, 6, 0},

{1, 0, 0, 1, 1, 0, 0, 1},

{0, 1, 0, 4, 0, 0, 1, 0},

{0, 0, 3, 1, 0, 0, 1, 1},

{1, 0, 6, 6, 0, 0, 1, 0},

{1, 0, 5, 0, 1, 1, 0, 6}

};

int linhaEscolhida, colunaEscolhida; // variaveis para as escolhas serem preenchidas

int escolha; // variavel para o usuario digitar a escolha

int YesOrNo; // variavel para o usuario ter opção de continuar

system("cls");

printf("Escolha a posição das peças do seu tabulheiro!\n\n"); // printf para orientar o usuário

for (L = 0; L < 8; L++)

{

for (C = 0; C < 8; C++)

{

printf("%d ", tabuleiroEscolhido[L][C]);

}

printf("\n");

}

printf("\nDigite algum número da linha do 1 até o 8 que deseja alterar: \n\n");

scanf\_s("%d", &linhaEscolhida);

while ((linhaEscolhida < 1) || (linhaEscolhida > 8)) // inserindo as escolhas do numero 1 ao 8

{

printf("\nO valor inserido está incorreto, deu erro! Digite novamente algum número do 1 ao 8 para a posição na linha!\n\n");

scanf\_s("%d", &linhaEscolhida);

}

printf("\nDigite algum número da coluna do 1 até o 8 que deseja alterar: \n\n");

scanf\_s("%d", &colunaEscolhida);

while ((colunaEscolhida < 1) || (colunaEscolhida > 8))

{

printf("\nO valor inserido está incorreto, deu erro! Digite novamente algum número do 1 ao 8 para a posição na coluna!\n\n");

scanf\_s("%d", &colunaEscolhida);

}

printf("\nDigite o número da peça que deseja inserir na posição\n");

printf("\n1 = Peão / 2 = Cavalo / 3 = Torre / 4 = Bispo / 5 = Rei / 6 = Rainha \n\n", linhaEscolhida, colunaEscolhida);

printf("1 = Peão\n");

printf("2 = Cavalo\n");

printf("3 = Torre\n");

printf("4 = Bispo\n");

printf("5 = Rei\n");

printf("6 = Rainha\n\n");

scanf\_s("%d", &escolha);

while ((escolha < 0) || (escolha > 6)) //condição de escolha do usuário

{

printf("\nA peça escolhida não existe! O programa deu erro, escolha outro numero correto de acordo as peças\n\n");

scanf\_s("%d", &escolha); // caso a condição não é cumprida imprimir isso

}

tabuleiroEscolhido[linhaEscolhida - 1][colunaEscolhida - 1] = escolha;

/\*diminindo um numero para que a escolha na matriz faça sentido, por que a sequencia na matriz sempre começa pelo zero.

Então através disso, se o usuário escolher 1 vai ser 1.

\*/

printf("\nDeseja fazer outra alteração? Se (SIM) digite 1, caso contrário digite 2: \n\n");

scanf\_s("%d", &YesOrNo);

while (YesOrNo == 1) // condição de escolha do usuário sim ou nao

{

printf("\nDigite a posição da linha do 1 até o 8 que deseja alterar:\n\n");// se sim imprimir isso

scanf\_s("%d", &linhaEscolhida);

while ((linhaEscolhida < 1) || (linhaEscolhida > 8)) //condição de escolha da linha

{

printf("\nEsta posição não existe, deu erro! Digite um número de 1 até o 8 para a posição na linha!\n\n");

scanf\_s("%d", &linhaEscolhida); // condição de escolha inexistente

}

printf("\nAgora, digite a posição da coluna do 1 até o 8 que deseja alterar:\n\n");

scanf\_s("%d", &colunaEscolhida);

while ((colunaEscolhida < 1) || (colunaEscolhida > 8)) // condição de escolha

{

printf("\nEsta posição não existe, deu erro! Digite um número de 1 a 8 para a posição na coluna!\n\n");

scanf\_s("%d", &colunaEscolhida);

}

printf("\nDigite a peça que deseja inserir na posição escolhida da linha %d e da coluna %d:\n\n", linhaEscolhida, colunaEscolhida);

printf("1 = Peão\n");

printf("2 = Cavalo\n");

printf("3 = Torre\n");

printf("4 = Bispo\n");

printf("5 = Rei\n");

printf("6 = Rainha\n\n");

scanf\_s("%d", &escolha);

printf("O tabulheiro escolhido ficou desta forma!\n\n");

int linhaEscolhida1, colunaEscolhida1;

for (linhaEscolhida1 = 0; linhaEscolhida1 < 8; linhaEscolhida1++)

{

for (colunaEscolhida1 = 0; colunaEscolhida1 < 8; colunaEscolhida1++)

{

printf("%d ", tabuleiroEscolhido[linhaEscolhida1][colunaEscolhida1]);

}

printf("\n");

}

int e1, e2;

for (e1 = 0; e1 < 8; e1++) //Resultado das escolhas feitas

{

for (e2 = 0; e2 < 8; e2++)

{

switch (tabuleiroEscolhido[e1][e2]) //montando casos das escolhas

{

case vazio:

cont[0]++;

break;

case peao:

cont[1]++;

break;

case cavalo:

cont[2]++;

break;

case torre:

cont[3]++;

break;

case bispo:

cont[4]++;

break;

case rei:

cont[5]++;

break;

case rainha:

cont[6]++;

break;

default:

break;

}

}

}

printf("\n");

printf("Cada número abaixo representa uma peça.\n\n");

printf("1 = Peão\n2 = Cavalo\n3 = Torre\n4 = Bispo\n5 = Rei\n6 = Rainha \n\n");

printf("Aonde tem o ZERO, significa que ali é ausente de peça!\n\n");

printf("Quantidade de peças:\n\n");

printf("Total de Peoes = %d\n", cont[1]);

printf("Total de Cavalos = %d\n", cont[2]);

printf("Total de Torres = %d\n", cont[3]);

printf("Total de Bispos = %d\n", cont[4]);

printf("Total de Reis = %d\n", cont[5]);

printf("Total de Rainhas = %d\n", cont[6]);

printf("Total geral das peças é = %d\n", cont[1]+cont[2]+cont[3]+cont[4]+cont[5]+cont[6]);

while ((escolha < 0) || (escolha > 6)) // condição de escolha de peças

{

printf("\nA peça que escolheu não existe! Digite outro numero de acordo com a tabela!\n\n");

printf("1 = Peão\n");

printf("2 = Cavalo\n");

printf("3 = Torre\n");

printf("4 = Bispo\n");

printf("5 = Rei\n");

printf("6 = Rainha\n\n");

scanf\_s("%d", &escolha);

}

tabuleiroEscolhido[linhaEscolhida - 1][colunaEscolhida - 1] = escolha;

printf("\nDeseja fazer outra alteração? Digite 1 para SIM e 2 para NÃO\n");

scanf\_s("%d", &YesOrNo);

}

printf("O tabulheiro escolhido ficou desta forma!\n\n");

int linhaEscolhida1, colunaEscolhida1;

for (linhaEscolhida1 = 0; linhaEscolhida1 < 8; linhaEscolhida1++)

{

for (colunaEscolhida1 = 0; colunaEscolhida1 < 8; colunaEscolhida1++)

{

printf("%d ", tabuleiroEscolhido[linhaEscolhida1][colunaEscolhida1]);

}

printf("\n");

}

int e1, e2;

for (e1 = 0; e1 < 8; e1++) //Resultado das escolhas feitas

{

for (e2 = 0; e2 < 8; e2++)

{

switch (tabuleiroEscolhido[e1][e2]) // montando estrutura dos casos das escolhas feitas, para então ser imprimida o resultado

{

case vazio:

cont[0]++;

break;

case peao:

cont[1]++;

break;

case cavalo:

cont[2]++;

break;

case torre:

cont[3]++;

break;

case bispo:

cont[4]++;

break;

case rei:

cont[5]++;

break;

case rainha:

cont[6]++;

break;

default:

break;

}

}

}

printf("\n");

printf("Cada número abaixo representa uma peça.\n\n");

printf("1 = Peão\n2 = Cavalo\n3 = Torre\n4 = Bispo\n5 = Rei\n6 = Rainha \n\n");

printf("Aonde tem o ZERO, significa que ali é ausente de peça!\n\n");

printf("Quantidade de peças:\n\n");

printf("Total de Peoes = %d\n", cont[1]);

printf("Total de Cavalos = %d\n", cont[2]);

printf("Total de Torres = %d\n", cont[3]);

printf("Total de Bispos = %d\n", cont[4]);

printf("Total de Reis = %d\n", cont[5]);

printf("Total de Rainhas = %d\n", cont[6]);

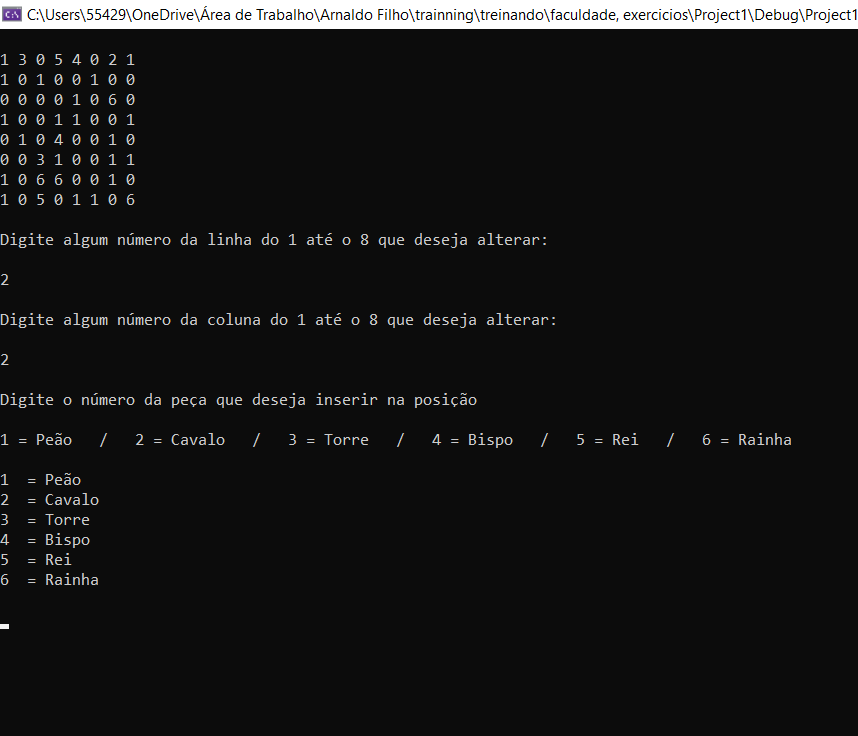
printf("Total geral das peças é = %d\n", cont[1]+cont[2]+cont[3]+cont[4]+cont[5]+cont[6]);//resultado geral

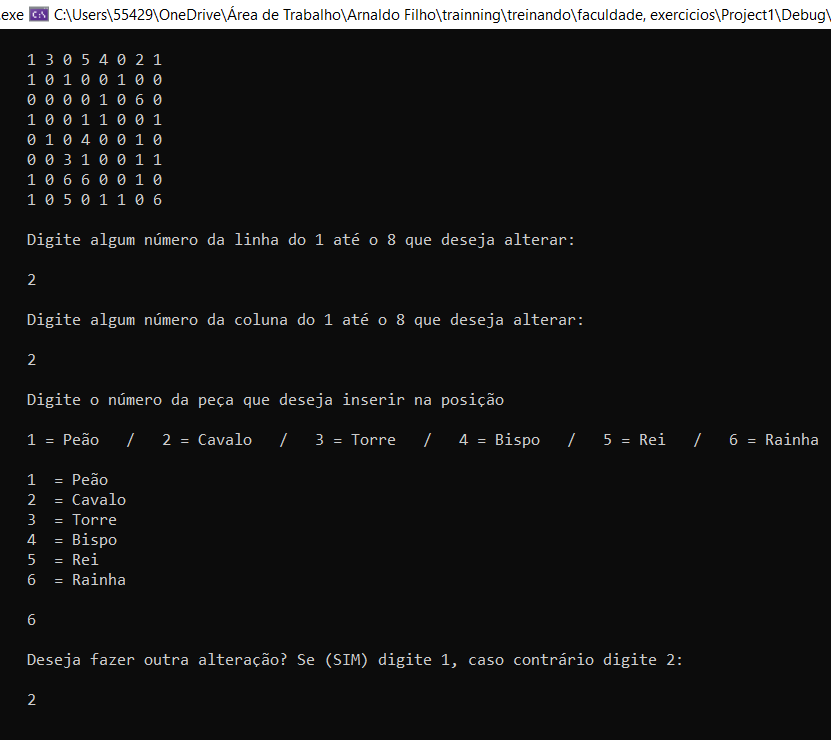
printf("\n");

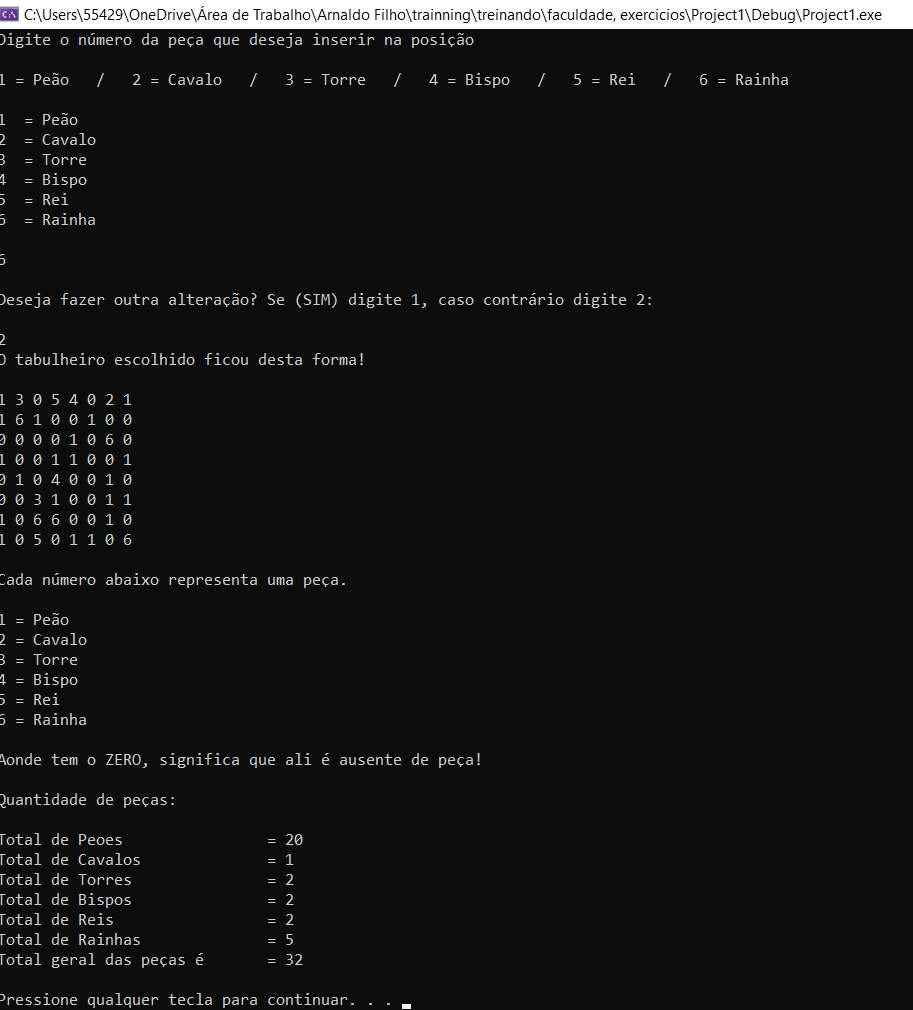
system("pause");

return(0);

}





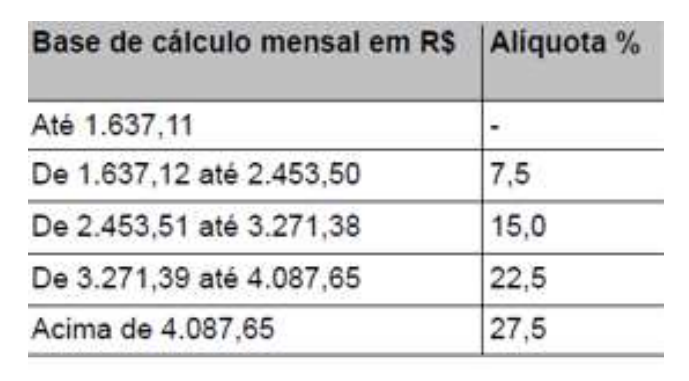


# **2 exercício**

ENUNCIADO: Escreva um programa, em LINGUAGEM C, que:

a) Solicite ao usuário a entrada de um registro que armazene na memória o Nome do funcionário e o seu salário.

b) Utilizando o salário informado calcule o imposto de renda devido e mostre na tela, conforme a tabela abaixo:



Solução do aluno:

**Não esqueça de explicar o código com comentários!!**

**CTRL+C/CTRL+V do Visual Studio!!**

**Se desejar, defina as páginas do documento para PAISAGEM para melhor inserir os códigos.**

**#include** <stdio.h>

#include <stdlib.h>

#include <locale.h>

#include <string.h>

typedef struct

{

float salario;

char nome[100]; //estrutura de dados para acessar o perfil do funcionario

}funcionario;

funcionario Funcionario;

int main()

{

setlocale(LC\_ALL, "Portuguese"); // com isso, ele faz funcionar todas as funções da lingua portuguesa, assim ele aceita todas acentuações

printf("\n\nOlá! Meu nome é Arnaldo Rocha Filho, sou aluno da UNINTER, do curso de engenharia da computação.\n\n");

system("pause");

system("cls"); //limpa a janela para ficar mais bonito

printf("\n\nAtravés deste programa vamos ver o imposto de renda do seu salário!\n\n");

system("pause");

system("cls");

strcpy\_s(Funcionario.nome, "NULL"); //deixa o campo limpo para que possa inserir os dados

Funcionario.salario = 0;

printf("\n\nQual é o seu nome? Digite aqui: ");

gets\_s(Funcionario.nome); //le o que o usuario digita para que seja imprimida na tela o que ele digitar

system("pause");

system("cls");

printf("\n\nQual é seu salário? Digite aqui: ");

scanf\_s("%f", &Funcionario.salario); //le o que o usuario digita para que seja imprimida na tela o que ele digitar

system("pause");

system("cls");

printf("\n\nO nome cadastrado ficou %s e seu salário é de R$%.2f\n\n", Funcionario.nome, Funcionario.salario); //imprime o resultado do que o usuario digitou

system("pause");

system("cls");

float imposto1, imposto2, imposto3, imposto4; //variaveis para fazer o calculo do imposto de renda

float imp1, imp2, imp3, imp4;

if (Funcionario.salario <= 1637.11) //imprimi na tela se o salario do usuario corresponder a condição mencionada

{

printf("\n\nO salario permanece com R$%.2f, pois ele é isento de imposto. \n\n ", Funcionario.salario);

}

else if (Funcionario.salario > 1637.11 || Funcionario.salario <= 2453.50) //imprimi na tela se o salario do usuario corresponder a condição mencionada

{

imposto1 = Funcionario.salario \* 0.075; // calculo do imposto

imp1 = Funcionario.salario - imposto1; //calculo do salario que vai ficar com o imposto

printf("\n\nSeu salario com desconto do imposto fica R$%.2f \n\n", imp1);

}

else if (Funcionario.salario > 2453.50 || Funcionario.salario <= 3271.38) //imprimi na tela se o salario do usuario corresponder a condição mencionada

{

imposto2 = Funcionario.salario \* 0.15;

imp2 = Funcionario.salario - imposto2;

printf("\n\nSeu salario com desconto do imposto fica R$%.2f \n\n", imp2);

}

else if (Funcionario.salario > 3271.38 || Funcionario.salario <= 4087.65) //imprimi na tela se o salario do usuario corresponder a condição mencionada

{

imposto3 = Funcionario.salario \* 0.22;

imp3 = Funcionario.salario - imposto3;

printf("\n\nSeu salario com desconto do imposto fica R$%.2f \n\n", imp3);

}

else if (Funcionario.salario > 4087.65)

{

imposto4 = Funcionario.salario \* 0.27;

imp4 = Funcionario.salario - imposto4;

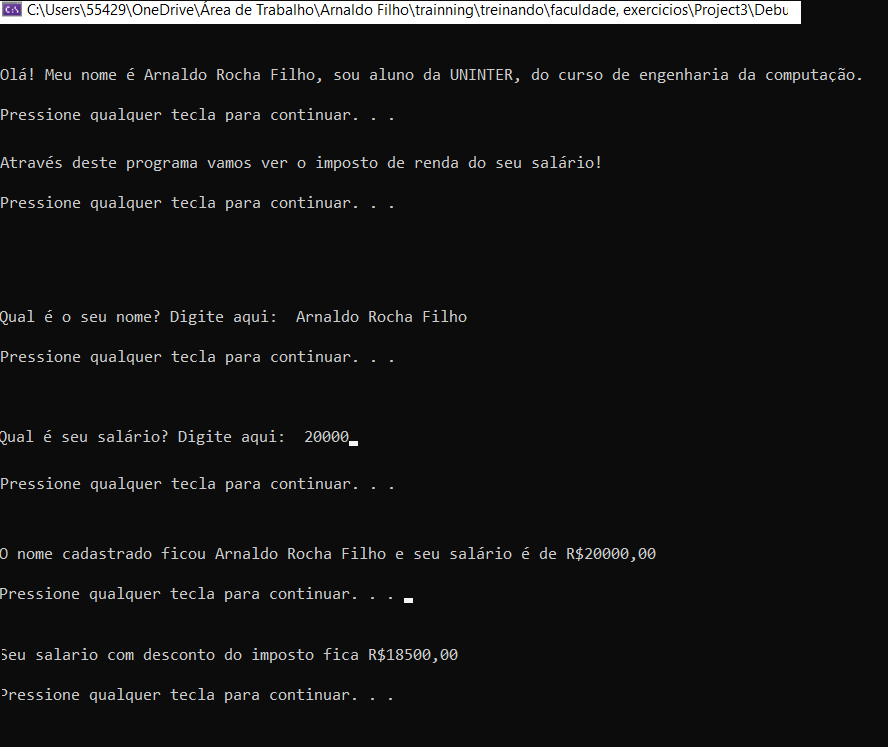
printf("\n\nSeu salario com desconto do imposto fica R$%.2f \n\n ", imp4);

}

system("pause");

return(0);

}



# **3 exercício**

ENUNCIADO: Faça um programa, em linguagem C, que lê 5 frases, informadas pelo usuário, de, no máximo, 50 caracteres cada uma e armazene-as em um arquivo. Mas, antes de gravar cada frase no arquivo, é necessário converter todas as suas letras para maiúsculas.

O nome do arquivo será fornecido, via teclado, pelo usuário.

A seguir, feche o arquivo.

Reabra o arquivo, para leitura, exibindo na tela todas as frases convertidas.

|  |
| --- |
| Dica   1. A função que converte minúscula para maiúscula é toupper(). 2. Como cada texto pode ter tamanho diferente, será necessário gravar antes de cada frase o tamanho do texto a ser lido. Logo serão necessários dois comandos de gravação e leitura (um para o número inteiro que indica a quantidade de caracteres de frase e outro para a frase com o tamanho lido. |

Solução do aluno:

**Não esqueça de explicar o código com comentários!!**

**CTRL+C/CTRL+V do Visual Studio!!**

**Se desejar, defina as páginas do documento para PAISAGEM para melhor inserir os códigos.**

**#include** <stdio.h>

#include <stdlib.h>

#include <locale.h>

#include <string.h>

#include <ctype.h> //biblioteca para acessar o toupper

int main()

{

setlocale(LC\_ALL, "Portuguese"); // com isso, ele faz funcionar todas as funções da lingua portuguesa, assim ele aceita todas acentuações

printf("\n\n\nOlá! Meu nome é Arnaldo Rocha Filho, sou aluno da UNINTER, do curso de engenharia da computação.\n\n\n");

system("pause");

system("cls"); //limpa a janela para ficar mais bonito

printf("\n\n\nNeste programa vamos converter 5 frases informadas pelo usuário em letra maíuscula.\n\n\n ");

system("pause");

system("cls");

char letras[50], frase1[50], frase2[50], frase3[50], frase4[50], frase5[50]; //variaveis string para ler e imprimir as frases

int tamanho; //variavel inteira para acessar a matriz

printf("\n\n\nDigite a primeira frase para ser convertida em letra maíscula: ");

gets\_s(letras);

for (tamanho = 0; tamanho < 50; tamanho++)

{

frase1[tamanho]=toupper(letras[tamanho]); //laço de repetição para acessar todas as letras da string

}

printf("\n\n\nSua primeira frase em letra maíscula fica: %s\n\n\n", frase1); //imprime o resultado

system("pause");

system("cls");

printf("\n\n\nDigite a segunda frase para ser convertida em letra maíscula: ");

gets\_s(letras);

for (tamanho = 0; tamanho < 50; tamanho++)

{

frase2[tamanho] = toupper(letras[tamanho]);

}

printf("\n\n\nSua segunda frase em letra maíscula fica: %s\n\n\n", frase2);

system("pause");

system("cls");

printf("\n\n\nDigite a terceira frase para ser convertida em letra maíscula: ");

gets\_s(letras);

for (tamanho = 0; tamanho < 50; tamanho++)

{

frase3[tamanho] = toupper(letras[tamanho]);

}

printf("\n\n\nSua terceira frase em letra maíscula fica: %s\n\n\n", frase3);

system("pause");

system("cls");

printf("\n\n\nDigite a quarta frase para ser convertida em letra maíscula: ");

gets\_s(letras);

for (tamanho = 0; tamanho < 50; tamanho++)

{

frase4[tamanho] = toupper(letras[tamanho]);

}

printf("\n\n\nSua quarta frase em letra maíscula fica: %s\n\n\n", frase4);

system("pause");

system("cls");

printf("\n\n\nDigite a quinta frase para ser convertida em letra maíscula: ");

gets\_s(letras);

for (tamanho = 0; tamanho < 50; tamanho++)

{

frase5[tamanho] = toupper(letras[tamanho]);

}

printf("\n\n\nSua quinta frase em letra maíscula fica: %s\n\n\n", frase5);

system("pause");

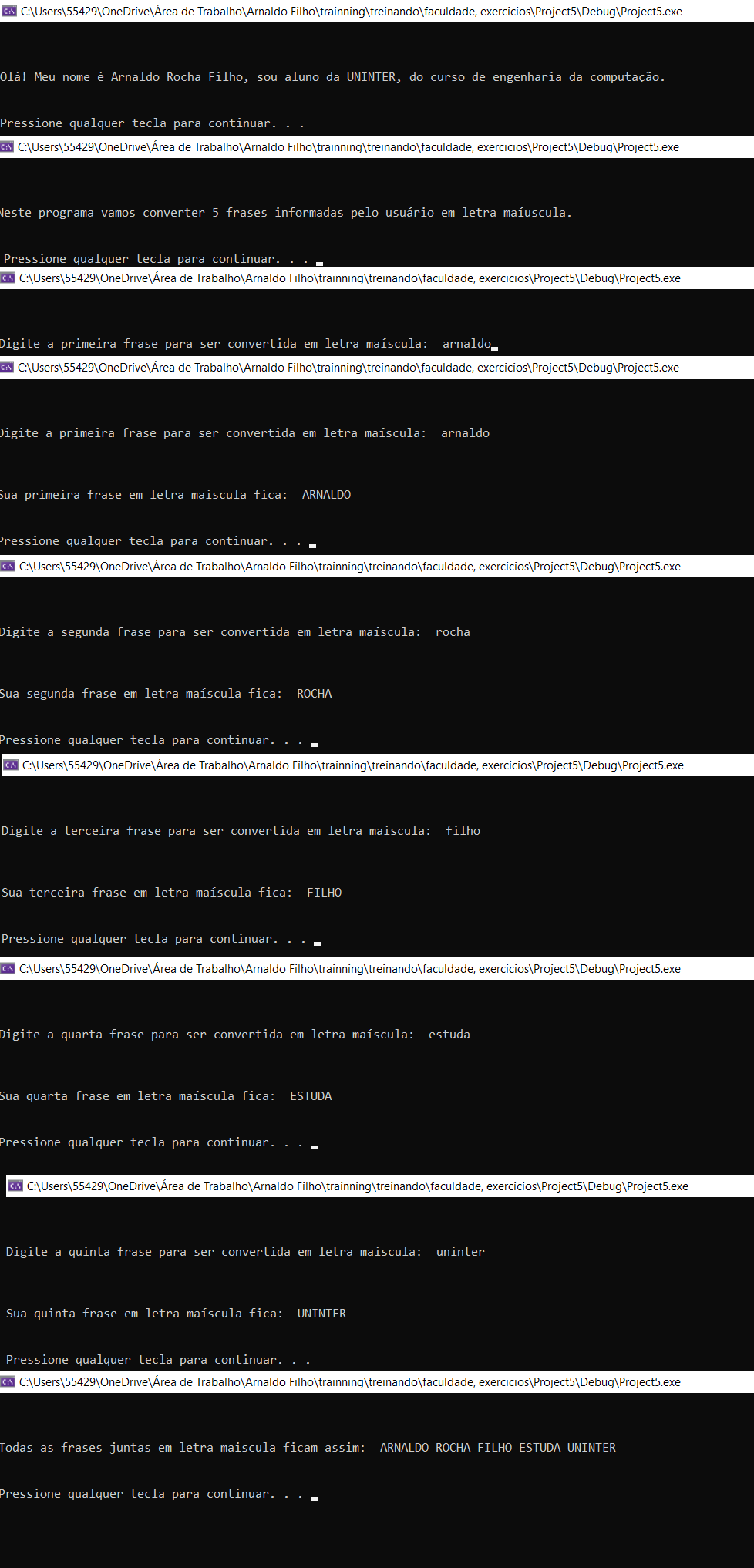
system("cls");

printf("\n\n\nTodas as frases juntas em letra maiscula ficam assim: %s %s %s %s %s\n\n\n", frase1, frase2, frase3, frase4, frase5); //imprime o resultado

system("pause");

return(0);

}

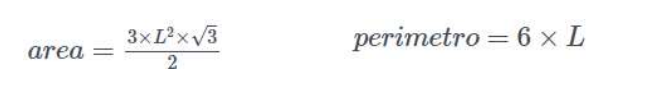


**4 EXERCÍCIO**

ENUNCIADO: Faça um programa, em linguagem C, para calcular a área e o perímetro de um hexágono. O programa deve implantar uma função chamada calc\_hexa que calcula a área e o perímetro de um hexágono regular de lado L. O programa deve solicitar ao usuário o lado do polígono, calcular e imprimir a área e o perímetro do polígono. O programa termina quando for digitado um valor negativo qualquer para o lado. A função deve obedecer ao seguinte protótipo:

**void calc\_hexa(float L, float\*area, float\*perimetro);**

Lembrando que a área e o perímetro de um hexágono regular são dados por:



Para os cálculos, obrigatoriamente você deve utilizar as funções sqrt e pow da biblioteca math.h

Solução do aluno:

**Não esqueça de explicar o código com comentários!!**

**CTRL+C/CTRL+V do Visual Studio!!**

**Se desejar, defina as páginas do documento para PAISAGEM para melhor inserir os códigos.**

**#include** <stdio.h>

#include <locale.h>

#include <stdlib.h>

#include <math.h>

void calc\_hexa(float L, float\* area, float\* perimetro) { //protótipo da função

\*area = 3 \* pow(L, 2) \* sqrt(3) / 2; //formula para ser possivel o calculo

\*perimetro = 6 \* L;

}

int main()

{

setlocale(LC\_ALL, "Portuguese"); // com isso, ele faz funcionar todas as funções da lingua portuguesa, assim ele aceita todas acentuações

printf("\n\nOlá! Meu nome é Arnaldo Rocha Filho, sou aluno da UNINTER, do curso de engenharia da computação.\n\n\n");

system("pause");

system("cls"); //limpa a janela para ficar mais bonito

printf("\n\nEntão bora calcular a área e o perímetro de um hexágono\n\n\n");

system("pause");

system("cls");

float L, perimetro, area; //variaveis para ser possivel o calculo

printf\_s("\n\nPara calcular a área e o perímitro do hexágono, digite o lado que voce quer aqui: ");

scanf\_s("%f", &L);

system("pause");

system("cls");

if (L <= -1) //condição se digitar valor negativo o programa fecha.

{

printf\_s("\n\nPara que funcione digite um valor positivo!\n\n");

}

else

{

calc\_hexa(L, &area, &perimetro); //calculo do hexagono atraves do lado escolhido

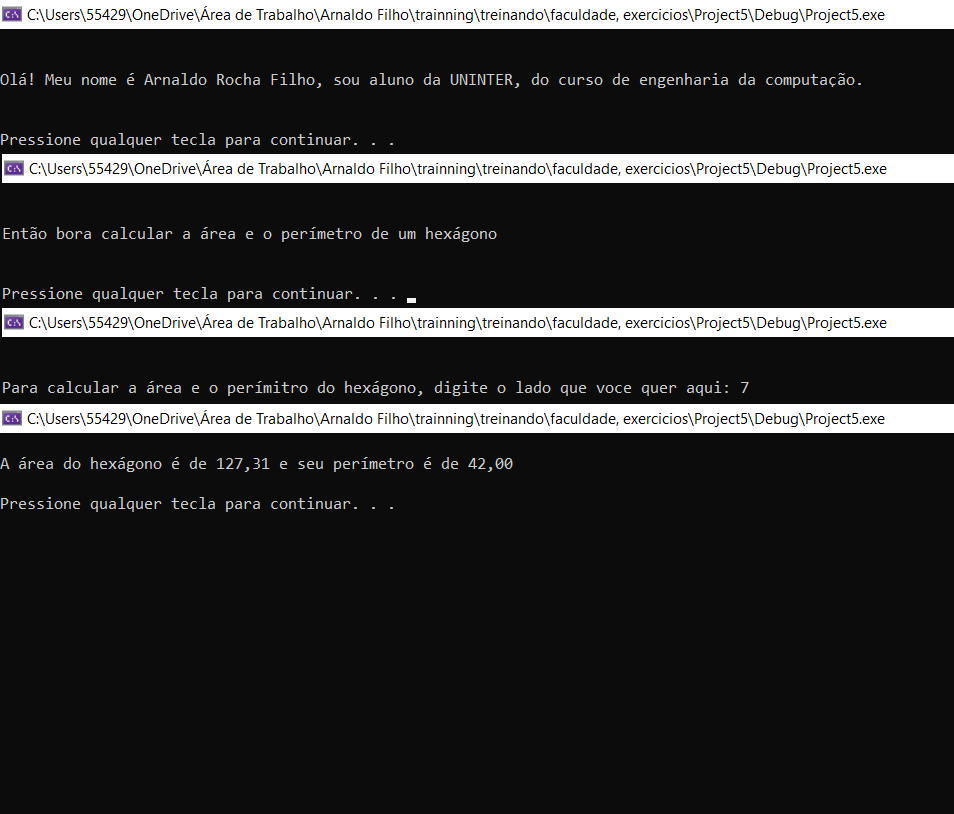
printf\_s("\nA área do hexágono é de %.2f e seu perímetro é de %.2f\n\n", area, perimetro); //imprimi o resultado

}

system("pause");

return 0;

}



**5 EXERCÍCIO**

ENUNCIADO: Faça um programa, em linguagem C, que:

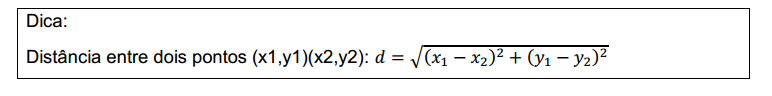
1. Crie um struct chamada ponto2d que tenha como atributos os pontos x,y.
2. Crie duas estruturas do tipo ponto2d chamadas ponto\_inicial e ponto\_final
3. Mostre um menu com as seguintes opções e implemente-as:

[1] - Digitar os valores do primeiro ponto

[2] - Digitar os valores do segundo ponto

[3] - Mostrar a distância entre os pontos

[4] - Sair



Solução do aluno:

**Não esqueça de explicar o código com comentários!!**

**CTRL+C/CTRL+V do Visual Studio!!**

**Se desejar, defina as páginas do documento para PAISAGEM para melhor inserir os códigos.**

**#include** <stdio.h>

#include <locale.h>

#include <stdlib.h>

#include <math.h> //biblioteca para funcionar o pow e sqrt, necessários para realizar calculos matematicos

int main()

{

setlocale(LC\_ALL, "Portuguese"); // com isso, ele faz funcionar todas as funções da lingua portuguesa, assim ele aceita todas acentuações

printf("\n\nOlá! Meu nome é Arnaldo Rocha Filho, sou aluno da UNINTER, do curso de engenharia da computação.\n\n\n");

system("pause");

system("cls"); //limpa a janela para ficar mais bonito

int opcoes = 0; //variavel com o valor zerado para ser preenchido

float d\_pontos = 0; //variavel com o valor zerado para ser preenchido

struct ponto2d //estrutura de dados para ver distancia dos pontos, de acordo com valor inserido pelo usuário

{

float x = 0, y = 0; //ponto x e ponto y

};

struct ponto\_inicial //estrutura de dados do ponto incial escolhido

{

}; struct ponto2d distancia\_inicial;

struct ponto\_final //estrutura de dados do ponto final escolhido

{

}; struct ponto2d distancia\_final;

while (opcoes > 0 || opcoes < 5) /\*condição de escolha dos casos abaixo do 1 até o 4, com loop,

caso digite um numero incorreto, vai ser exibida uma mensagem para digitar novamente

\*/

{

printf("\n\n[1] - Digitar os valores do primeiro ponto.\n");

printf("[2] - Digitar os valores do segundo ponto.\n");

printf("[3] - Mostrar a distancia entre os dois pontos.\n");

printf("[4] - Sair.\n\n");

scanf\_s("%d", &opcoes);

system("pause");

system("cls");

switch (opcoes)

{

case 1:

printf("\n\nFoi escolhido a opção [1] da tabela, então....");

printf("\nDigite o primeiro número do ponto 1: ");

scanf\_s("%f", &distancia\_inicial.x);//reconhece o numero digitado pelo usuario

printf("\n\nDigite o segundo número do ponto 1: ");

scanf\_s("%f", &distancia\_inicial.y); //reconhece o numero digitado pelo usuario

system("pause");

system("cls");

break;

case 2:

printf("\n\nFoi escolhido a opção [2] da tabela, então....");

printf("\n\nDigite o primeiro número do ponto 2: ");

scanf\_s("%f", &distancia\_final.x);

printf("\nDigite o segundo número do ponto 2: ");

scanf\_s("%f", &distancia\_final.y);

system("pause");

system("cls");

break;

case 3:

d\_pontos = pow((distancia\_inicial.x - distancia\_final.x), 2) + pow((distancia\_inicial.y - distancia\_final.y), 2); //calcula a distancia dos pontos

d\_pontos = sqrt(d\_pontos);

printf("\n\nFoi escolhida a opção [3] da tabela, que vê o resultado da distância entre os pontos escolhidos nas opções anteriores.");

printf("\nA distância entre os pontos é %.2f\n\n\n", d\_pontos);

system("pause");

system("cls");

break;

case 4:

system("pause");

return 0; //se o usuário escolher a opção 4 o programa para

break;

default: printf\_s("\n\nDigite as entrada corretamente, de acordo como na tabela!\n\n");

system("pause");

system("cls");

}

}

system("pause");

system("cls");

}

