1 - Faça um algoritmo que leia dados inteiros e armazene-os em uma matriz 3x4. Em seguida, mostre a quantidade de números pares e ímpares armazenados na matriz.

#include <stdio.h>

int main(){

int e[3][4],lin,col,impar=0,par=0;

for (lin=0;lin<3;lin++){

for (col=0; col<4; col++){

scanf ("%d", &e[lin][col]);

if (e[lin][col]%2==0){

par+=1;

}else{

impar+=1;

}

}

}

printf ("\n%d números pares", par);

printf ("\n%d números impares", impar);

printf ("\nSOMENTE OS PARES");

for (lin=0; lin<3; lin++){

printf ("\n");

for (col=0; col<4; col++){

if (e[lin][col]%2==0){

printf ("%d ", e[lin][col]);

}else{

printf("X ");

}

}

}

printf ("\nSOMENTE OS IMPARES");

for (lin=0; lin<3; lin++){

printf ("\n");

for (col=0; col<4; col++){

if (e[lin][col]%2!=0){

printf ("%d ", e[lin][col]);

}else{

printf("X ");

}

}

}

}

2 - Faça um algoritmo que leia números inteiros e armazene-os na matriz 4x4. Porém, na diagonal principal, não armazene o número lido, e sim um 0 (zero).

Na diagonal principal, os elementos têm linha = colona: [0][0], [1][1], [2][2], [3][3]

#include <stdio.h>

int main(){

int e[4][4],lin,col;

for (lin=0;lin<4;lin++){

for (col=0; col<4; col++){

if (lin==col){

e[lin][col] = 0;

}else{

scanf ("%d", &e[lin][col]);

}

}

}

for (lin=0; lin<4; lin++){

printf("\n");

for (col=0; col<4; col++){

printf ("%d ", e[lin][col]);

}

}

}

3 - Faça um algoritmo que leia uma matriz 4x4 de números inteiros. Gere uma segunda matriz, na qual as linhas são as colunas da matriz 1, e as colunas são as linhas da matriz 1.

#include <stdio.h>

int main(){

int e[4][4],lin,col;

for (lin=0;lin<4;lin++){

for (col=0; col<4; col++){

scanf ("%d", &e[lin][col]);

}

}

printf("\nMATRIZ ATUAL\n");

for (lin=0;lin<4;lin++){

printf("\n");

for (col=0; col<4; col++){

printf ("%d ", e[lin][col]);

}

}

printf("\n\nMATRIZ INVERSA\n");

for (col=0; col<4; col++){

printf("\n");

for (lin=0; lin<4; lin++){

printf ("%d ", e[lin][col]);

}

}

}

4 - Faça um algoritmo que leia dados e armazene em uma matriz 3x3 de números inteiros. Em seguida, mostre os elementos que sejam iguais ao maior número armazenado na matriz.

#include <stdio.h>

int main(){

int e[3][3],lin,col,maior=0,repeticao=0;

for (lin=0;lin<3;lin++){

for (col=0; col<3; col++){

scanf ("%d", &e[lin][col]);

if (e[lin][col]>maior){

maior = e[lin][col];

repeticao=1;

}else if (e[lin][col]==maior){

repeticao+=1;

}

}

}

for (lin=0;lin<3;lin++){

printf("\n");

for (col=0; col<3; col++){

if (e[lin][col]==maior){

printf ("%d ", e[lin][col]);

}else{

printf ("X ");

}

}

}

printf ("\n\nO maior número digitado foi o %d e repetiu %d vezes!",maior,repeticao);

}

5 - Faça um programa que gere uma matriz 5x5, conforme esta sequência:



int main()

{

int e[5][5],lin,col;

for (lin=0;lin<5;lin++){

for (col=0;col<5;col++){

if (lin==col){

e[lin][col]=0;

}else{

e[lin][col]=1;

}

}

}

for (lin=0;lin<5;lin++){

printf("\n");

for (col=0;col<5;col++){

printf ("%d ", e[lin][col]);

}

}

}