**import random**

**def crea\_matriz(renglones, columnas):**

**matriz = []**

**for i in range(renglones):**

**matriz.insert(i, [] )**

**for j in range(columnas):**

**valor = int(input("Introduce un valor: "))**

**matriz[i].insert(j, valor)**

**return matriz**

**def imprime\_matriz(m):**

**for i in range(len(m)):**

**for j in range(len(m[i])):**

**print("%4i" % m[i][j], end= ' ')**

**print()**

**def llena\_matriz(m):**

**for i in range(len(m)):**

**for j in range(len(m[i])):**

**m[i][j] = random.randint(-15, 20)**

**def suma\_positivos(m):**

**acum = 0**

**for i in range(len(m)):**

**for j in range(len(m[i])):**

**if m[i][j] > 0:**

**acum = acum + m[i][j]**

**return acum**

**def modifica\_celda(matriz, renglon, columna, valor):**

**matriz[renglon][columna] = valor**

**def suma\_diagonal\_inversa(m):**

**acum = 0**

**columna = len(m[0])-1**

**for i in range(len(m)):**

**if columna >= 0:**

**acum = acum + m[i][columna]**

**columna = columna-1**

**return acum**

**def suma\_diagonal\_inversa2(m):**

**acum = 0**

**columna = len(m[0])-1**

**for i in range(len(m)):**

**for j in range(len(m[i])):**

**if i+j == columna:**

**acum = acum + m[i][j]**

**return acum**

**def multiplica\_renglon(m, renglon, num):**

**for i in range(len(m)):**

**for j in range(len(m[i])):**

**if i == renglon:**

**m[i][j] = m[i][j] \* num**

**def intercambia\_renglones(m, r1, r2):**

**for i in range(len(m)):**

**for j in range(len(m[i])):**

**if i == r1:**

**temp = m[r1][j]**

**m[r1][j] = m[r2][j]**

**m[r2][j] = temp**

**def encuentraFamilia(matriz, cadena):**

**acum = 0**

**for i in range(0, len(matriz)):**

**for j in range(0, len(matriz[i])):**

**if cadena == matriz[i][j]:**

**return matriz[i]**

**break**

**return "el nombre no esta en la tabla"**

**"""def promedio(m):**

**acum = 0**

**for i in range(0, len(m)):**

**columnas = len(m[i])**

**for j in range(0, len(m[i])):**

**acum = acum + m[i][j]**

**return acum/ (len(m)\*columnas)**

**def multiplicaColumna(m, columna, num):**

**for i in range(0, len(m)):**

**for j in range(0, len(m[i])):**

**if j == columna:**

**m[i][j] = m[i][j] \* num**

**def posicionPares(m):**

**for i in range(0, len(m)):**

**for j in range(0, len(m[i])):**

**if m[i][j]%2==0:**

**print("El valor %i está en la posición %i, %i"%(m[i][j], i, j))**

**"""**

**def menu():**

**print()**

**print("1. Imprime matriz")**

**print("2. Llena matriz")**

**print("3. Suma positivos")**

**print("4. Modifica celda")**

**print("5. Suma diagonal inversa")**

**print("6. Multiplica renglón")**

**print("7. Intercambia renglones")**

**print("8. Encuentra nombres")**

**print("9. Salir")**

**#M = [ [1,2,3,4], [2,1,2,2], [3,2,1,2], [4,2,2,1] ] # Declarar una matriz**

**#M = [ [2,5,6,4], [3,4,5,1], [7,8,5,6], [9,7,1,5] ] # Declarar una matriz**

**nombresEpicos= [ ["Naofumi", "Filo", "Raphtalia"], ["Rand Al'thor", "Perrin Arabaya", "Mathrim Cauldron", "Egwene Al'vere", "Nynaieve Al'mere"], ["Lithany of Fury", "Macragge's Honour", "Vengeful Spirit", "Harbinger of Doom", "Chronicle of Ashes"], ["Cloud Strife", "Sephiroth", "Vincent Valentine", "Zack Fair", "Aerith Gainsborough", "Tifa Lockhart", "Barret Wallace", "Yuffie Kisaragi"], ["Cormyr", "WestGate", "Suzeil", "Menzoberranzan", "Waterdeep"], ["Atlas", "Dectective Comics", "Dark Horse", "Image"] ]**

**tabla\_periodica = [["litio", "sodio", "potasio", "rubidio", "cesio", "francio"], ["berilio", "magnesio", "calcio", "estroncio", "bario", "radio"], ["escandio", "itrio", "lantano", "actinio"], ["titanio", "circonio", "hafnio", "rutherfordio"], ["vanadio", "niobio", "tántalo", "dubnio"], ["cromo", "molibdeno", "wolframio", "seaborgio"], ["manganeso", "tecnecio", "renio"], ["hierro", "rutenio", "osmio", "hassio"], ["cobalto", "rodio", "Iridio", "meitnerio"], ["níquel", "paladio", "platino", "darmstadtio"], ["cobre", "plata", "oro"], ["zinc", "cadmio", "mercurio"], ["boro", "aluminio", "galio", "indio", "talio", "nihonio"], ["carbono", "silicio", "germanio", "estaño", "plomo", "flerovio"], ["nitrógeno", "fósforo", "arsénico", "antimonio", "bismuto", "moscovio"], ["oxígeno", "azufre", "selenio", "telurio", "polonio", "livermorio"], ["flúor", "cloro", "bromo", "yodo", "astato", "téneso"], ["helio", "neón", "argón", "kriptón", "xenón", "radón", "organesón"] ]**

**def main():**

**ren = int(input("Introduce el número de renglones: "))**

**col = int(input("Introduce el número de columnas: "))**

**M = crea\_matriz(ren, col)**

**continua = True**

**while continua:**

**menu()**

**opcion = int(input("Introduce una opcion: "))**

**if opcion == 1:**

**imprime\_matriz(M)**

**elif opcion == 2:**

**llena\_matriz(M)**

**imprime\_matriz(M)**

**elif opcion == 3:**

**imprime\_matriz(M)**

**res = suma\_positivos(M)**

**print("La suma de positivos es: %i" % res)**

**elif opcion == 4:**

**imprime\_matriz(M)**

**r = int(input("Introduce el número de renglón: "))**

**c = int(input("Introduce el número de columna: "))**

**num = int(input("Introduce el valor: "))**

**modifica\_celda(M, r, c, num)**

**imprime\_matriz(M)**

**elif opcion == 5:**

**imprime\_matriz(M)**

**res = suma\_diagonal\_inversa2(M)**

**print("La suma de la diagonal inversa es", res)**

**elif opcion == 6:**

**imprime\_matriz(M)**

**r = int(input("Introduce el número de renglón: "))**

**num = int(input("Introduce el número: "))**

**multiplica\_renglon(M, r, num)**

**imprime\_matriz(M)**

**elif opcion == 7:**

**imprime\_matriz(M)**

**r1 = int(input("Introduce el renglón 1: "))**

**r2 = int(input("Introduce el renglón 2: "))**

**intercambia\_renglones(M, r1, r2)**

**imprime\_matriz(M)**

**elif opcion == 8:**

**cadena = str(input("Introduce un elemento de la tabla periodica: "))**

**lista = encuentraFamilia(tabla\_periodica, cadena)**

**print(lista)**

**elif opcion == 9:**

**print("Adios")**

**continua = False**

**else:**

**print("Opcion\_invalida")**

**main()**