



Instituto Politécnico Nacional Escuela Superior de Cómputo



Paradigmas de Programación

Actividad: Práctica 2 de Laboratorio

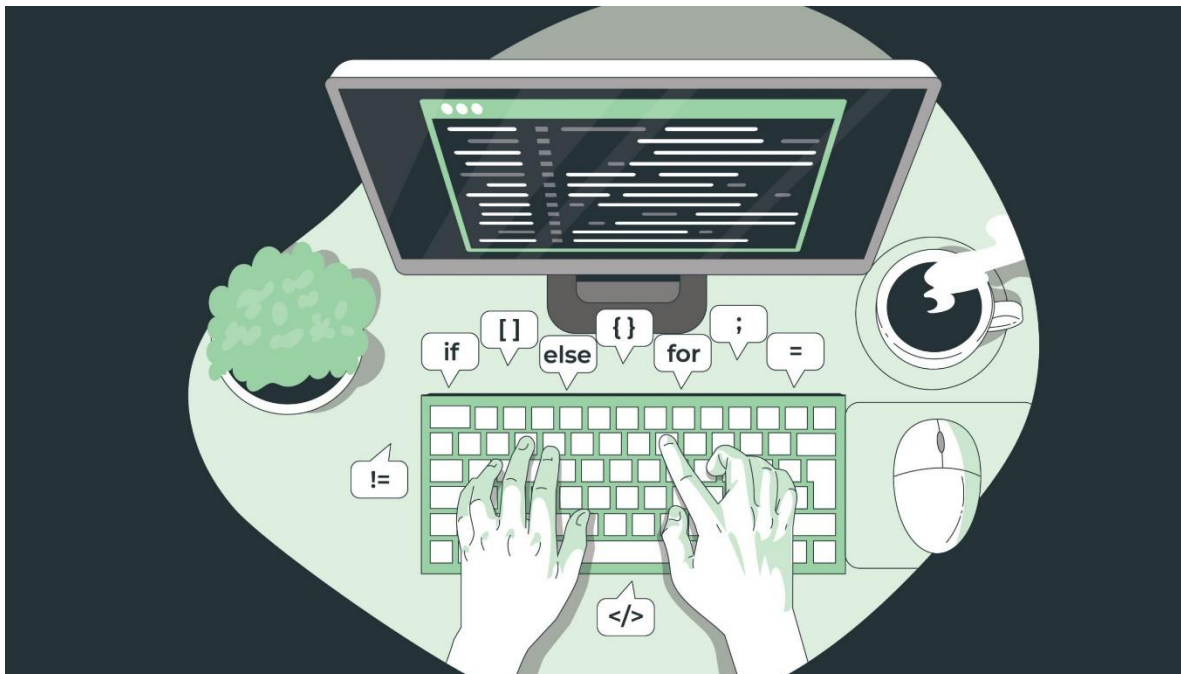
Profesor: García Floriano Andrés

Fecha: 12/03/2024

Alumno:

Pacheco Refugio Alan Ivan

Grupo: 3CV1



Objetivo

Migrar el código de un programa en Python a un lenguaje en este caso usé lenguaje C.

El programa en Python es la inserción de números en dos listas en donde se procede a buscar el máximo y el mínimo de cada lista, y sumar los elementos de ambas listas

Desarrollo

Este programa fue un poco confuso ya que trabajar o migrar un programa de un lenguaje a otro suele serlo, es por eso que al momento de visualizar las líneas de código del programa en Python pude apreciar que sus sentencias son muy particulares y bastante cortas a comparación del lenguaje C.

Código

Comencé declarando las librerías que iba a usar así como declarando el tamaño máximo de las listas para así tener una forma estática de ellas, hice uso de estructura de datos esto con la finalidad de poder usar los elementos las veces que sean necesarias. Posteriormente declaré los arreglos que usé inicializándolos en cero para que no guarde ningún valor sino hasta que el usuario teclee los que quiera

```
/* Pacheco Refugio Alan Ivan | Practica 2 Versión 2.0 | Paradigmas de Programación */  
  
#include<stdio.h>  
#include<math.h>  
#include<time.h>  
#include<string.h>  
  
#define MAX 10 // Tamaño de la Lista1 de numeros  
#define MAX1 10 // Tamaño de la Lista2 de numeros  
  
// Estructura de datos usada para la Lista1  
struct Lista1 {  
    int numero;  
};  
  
// Estructura de datos usada para la Lista2  
struct Lista2 {  
    int numero2;  
};  
  
// Declaración del arreglo para almacenar los Numeros en la Lista1  
struct Lista1 Lista1Numeros[MAX];  
int numElementos = 0; // Numero con el que comienza la Lista1  
  
// Declaración del arreglo para almacenar los Numeros en la Lista2  
struct Lista2 Lista2Numeros[MAX1];  
int numElementos2 = 0; // Numero con el que comienza la Lista2
```

Posteriormente usé funciones tipo void para tener un orden y para posteriormente hacer llamado a la función por completo haciendo uso de las estructuras de datos que antes había creado, estas funciones tienen el objetivo de que el usuario inserte los números que quiera guardándolos así en su respectiva lista y de esta manera guardarlos para las siguientes operaciones el uso de la sentencia if me sirvió de mucho para tener un orden de las cosas y darme cuenta de forma más rápida los posibles errores a la hora de depurar el programa. Estas funciones son para las dos listas, las cuales tienen su propia estructura de datos, las propias declaraciones de arreglos y variables.

```
// Función para insertar un Numero a la Lista1
void insertarNumero1() {
    if (numElementos < MAX) {
        struct Lista1 nuevoNumero;
        printf("Numero para Lista1: ");
        scanf("%d", &nuevoNumero.numero);

        Lista1Numeros[numElementos] = nuevoNumero;
        numElementos++;

        printf("Numero agregado correctamente a Lista1\n");
    }
    else {
        printf("La Lista1 está llena, no es posible agregar más Numeros :(\n");
    }
}

// Función para insertar un Numero a la Lista2
void insertarNumero2() {
    if (numElementos2 < MAX1) {
        struct Lista2 nuevoNumero2;
        printf("Numero para Lista2: ");
        scanf("%d", &nuevoNumero2.numero2);

        Lista2Numeros[numElementos2] = nuevoNumero2;
        numElementos2++;

        printf("Numero agregado correctamente a Lista2\n");
    }
    else {
        printf("La Lista2 está llena, no es posible agregar más Numeros :(\n");
    }
}
```

Continuando con el proceso hice uso de las funciones para así mostrar los datos de cada uno de las listas, de igual manera los números máximos y mínimos que se almacenan en cada una de ellas.

Hice uso de inteligencias artificiales y foros web en donde me ayude para saber de forma correcta la forma en que hay que ejecutar las funciones para máximos y mínimos.

```

void mostrarNumeros1() {
    printf("\nCantidad de Numeros en la Lista1: %d\n", numElementos);
    printf("Lista1 de Numeros: \n");
    for (int i = 0; i < numElementos; i++) {
        printf("Numero: %d\n", Lista1Numeros[i].numero);
    }
}

// Función para mostrar todos los Numeros en la Lista2
void mostrarNumeros2() {
    printf("\nCantidad de Numeros en la Lista2: %d\n", numElementos2);
    printf("Lista2 de Numeros: \n");
    for (int i = 0; i < numElementos2; i++) {
        printf("Numero: %d\n", Lista2Numeros[i].numero2);
    }
}

// Función para encontrar el Numero maximo en la Lista1
int encontrarMaximoLista1() {
    int maximo = Lista1Numeros[0].numero;
    for (int i = 1; i < numElementos; i++) {
        if (Lista1Numeros[i].numero > maximo) {
            maximo = Lista1Numeros[i].numero;
        }
    }
    return maximo;
}

```

Funcion hecha con ayuda de IAS y foros

Estas funciones de igual manera me ayuden de fuentes en internet

```

// Función para encontrar el Numero minimo en la Lista1
int encontrarMinimoLista1() {
    int minimo = Lista1Numeros[0].numero;
    for (int i = 1; i < numElementos; i++) {
        if (Lista1Numeros[i].numero < minimo) {
            minimo = Lista1Numeros[i].numero;
        }
    }
    return minimo;
}

// Función para encontrar el numero maximo en la Lista2
int encontrarMaximoLista2() {
    int maximo = Lista2Numeros[0].numero2;
    for (int i = 1; i < numElementos2; i++) {
        if (Lista2Numeros[i].numero2 > maximo) {
            maximo = Lista2Numeros[i].numero2;
        }
    }
    return maximo;
}

// Función para encontrar el numero minimo en la Lista2
int encontrarMinimoLista2() {
    int minimo = Lista2Numeros[0].numero2;
    for (int i = 1; i < numElementos2; i++) {
        if (Lista2Numeros[i].numero2 < minimo) {
            minimo = Lista2Numeros[i].numero2;
        }
    }
    return minimo;
}

```

Funcion hecha con ayuda de IAS y foros

Esta función hace la misión de sumar los elementos de cada una de las listas

```

// Función para calcular la suma de ambas listas
int sumaListas() {
    int suma = 0;
    for (int i = 0; i < numElementos; i++) {
        suma += Lista1Numeros[i].numero;
    }
    for (int i = 0; i < numElementos2; i++) {
        suma += Lista2Numeros[i].numero2;
    }
    return suma;
}

```

Todo esto se ve en un menú en donde cada opción esta asignada a una función haciendo uso del ciclo do while para repetir las veces que sean necesarias

```
int main () {
    int opcion;

    do{
        printf("\n\n ----- Menu ----- \n\n");
        printf("1) Insertar nuevo nmero en la Lista1\n");
        printf("2) Mostrar todos los elementos de la Lista1\n");
        printf("3) Insertar nuevo numero en la Lista2\n");
        printf("4) Mostrar todos los elementos de la Lista2\n");
        printf("5) Numero maximo y minimo de la Lista1\n");
        printf("6) Numero maximo y minimo de la Lista2\n");
        printf("7) Suma de ambas listas\n");
        printf("8) Tiempo de ejecucion del programa\n");
        printf("9) Salir\n");
        printf("\nSelecione una opcion por favor: ");
        scanf("\n%d", &opcion);

        switch (opcion){
            case 1:
                insertarNumero1();
                break;
            case 2:
                mostrarNumeros1();
                break;
            case 3:
                insertarNumero2();
                break;
            case 4:
                break;
            case 5:
                break;
            case 6:
                break;
            case 7:
                break;
            case 8:
                break;
            case 9:
                break;
            default:
                break;
        }
    } while (opcion != 9);

    return 0;
}
```