

Universidad Autónoma de Baja California



Ingeniería en Software y Tecnologías Emergentes.

Práctica 7. Punteros

Materia: Lenguaje C

Maestro: Yulith Altamirano

Alumno: Diego Quiros 372688

Fecha: 11/08/2023

Instrucciones:

Desarrollen el código en lenguaje C y elaboren el diagrama de flujo correspondiente para los ejercicios. Será suficiente con un archivo .cpp que contenga todos los ejercicios organizados en un menú.

Repositorio:

C/C++

https://github.com/diegovq12/LenguajeC-QVDD_932/tree/main/Practicas/Practica7

Problema 1 Argumentos en la Línea de Comandos

Crea un programa que tome dos números enteros como argumentos desde la línea de comandos.

Utiliza apuntadores para capturar estos argumentos y realizar las siguientes

operaciones:

- ✓ Suma los dos números y muestra el resultado.
- ✓ Resta el segundo número del primero y muestra el resultado.
- ✓ Multiplica los dos números y muestra el resultado.
- ✓ Divide el primer número por el segundo y muestra el resultado.

C/C++

```
int main(int argc, char *argv[])
{

    // convertimos los argumentos obtenidos a numeros
    int numero1 = atoi(argv[1]);
    int numero2 = atoi(argv[2]);

    // realizamos y mostramos el resultado
    printf("Suma: %d + %d = %d\n", numero1, numero2, numero1 + numero2);
    printf("Resta: %d - %d = %d\n", numero1, numero2, numero1 - numero2);
    printf("Multiplicacion: %d * %d = %d\n", numero1, numero2, numero1 *
numero2);
    if (numero2 != 0)
    {
        printf("Division: %d / %d = %.2f\n", numero1, numero2, (float)numero1 /
numero2);
    }
}
```

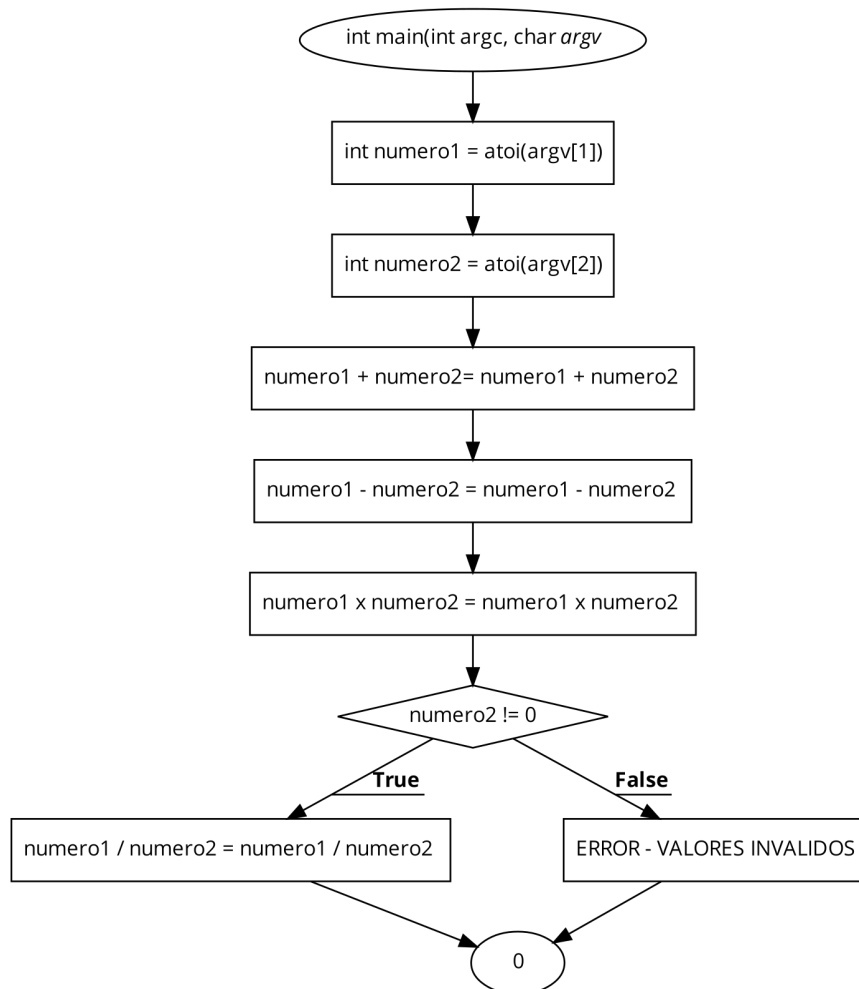
```

    }
    else
    {
        printf("ERROR - VALORES INVALIDOS");
    }

    return 0;
}

```

Diagrama de flujo



Problema 2 Ordenación de Palabras

✓ Crea un programa que incluya un arreglo de apuntadores a caracteres, donde cada apuntador apunta a una palabra (por ejemplo, "manzana", "banana", "cereza", "uva", "naranja").

- ✓ Utiliza apuntadores y la aritmética de direcciones para ordenar las palabras en orden alfabético.
- ✓ Imprime las palabras ordenadas en la consola.

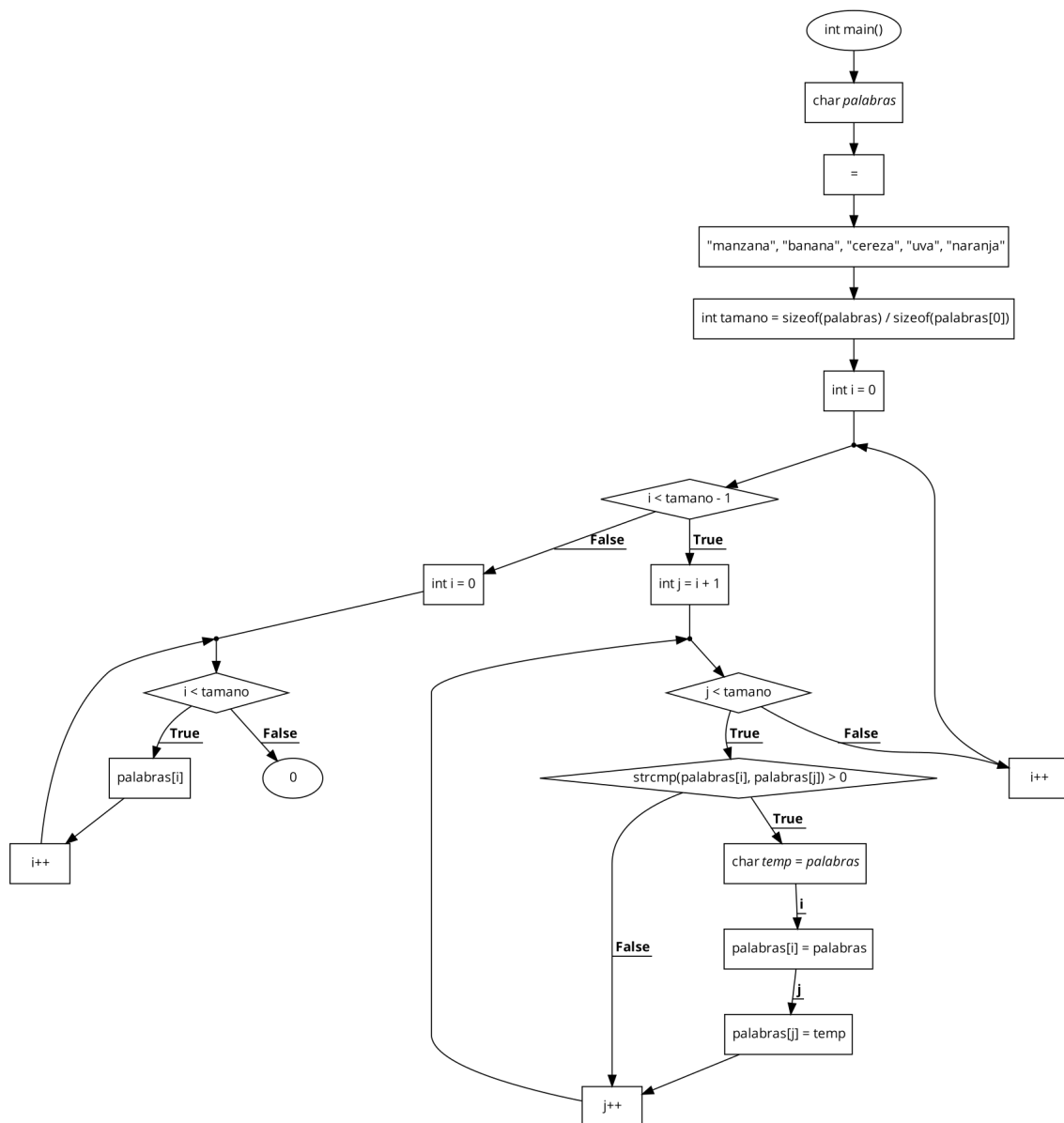
```
C/C++
int main()
{
    // Inicializamos el arreglo de palabras
    char *palabras[] = {"manzana", "banana", "cereza", "uva", "naranja"};
    // obtenemos el tamaño del vector
    int tamaño = sizeof(palabras) / sizeof(palabras[0]);

    // Ciclo para ordenar las palabras en orden alfabético
    for (int i = 0; i < tamaño - 1; i++)
    {
        for (int j = i + 1; j < tamaño; j++)
        {
            // Si la comparación entre ambas cadenas es mayor que cero significa que
            // no están en el orden correcto
            if (strcmp(palabras[i], palabras[j]) > 0)
            {
                // Ordenamos mediante el método de la burbuja
                char *temp = palabras[i];
                palabras[i] = palabras[j];
                palabras[j] = temp;
            }
        }
    }

    // Imprimir las palabras ordenadas
    for (int i = 0; i < tamaño; i++)
    {
        printf("%s\n", palabras[i]);
    }

    return 0;
}
```

Diagrama de flujo



Problema 3 Matriz Multidimensional

- Crea una matriz bidimensional de enteros, por ejemplo, 3x3, con algunos valores.
- Utiliza apuntadores y la aritmética de direcciones para recorrer la matriz y realizar las siguientes operaciones:
 - ✓ Calcula la suma de todos los elementos de la matriz.
 - ✓ Encuentra el elemento más grande en la matriz y muestra su valor y posición.

C/C++

```
int main()
{
    // Declaracion de una matriz 3x3
    int matriz[3][3] = {{1, 5, 3}, {4, 15, 6}, {1, 8, 9}};

    // Declarar variables para la suma y el elemento más grande
    int suma = 0;
    int elementoMayor = matriz[0][0];
    int filaElementoMayor = 0;
    int columnaElementoMayor = 0;

    int *ptr = &matriz[0][0]; // Inicializando el puntero en el primer elemento
    de la matriz

    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            // Sumar el elemento actual a la suma total
            suma += *ptr;

            // Comprobar si el elemento actual es mayor que el maximo encontrado
            hasta ahora
            if (*ptr > elementoMayor)
            {
                elementoMayor = *ptr;
                filaElementoMayor = i;
                columnaElementoMayor = j;
            }

            // Avanzar al siguiente elemento de la matriz
            ptr++;
        }
    }

    // Imprimir la matriz
    printf("Matriz 3x3:\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            printf("%d\t", matriz[i][j]);
        }
        printf("\n");
    }

    // Imprimir la suma ,el elemento mas grande y su posición
    printf("Suma de todos los elementos: %d\n", suma);
    printf("Elemento mas grande: %d\n", elementoMayor);
}
```

```

printf("El elemento mas grande se encuentra en la posicion [%d][%d]\n",
filaElementoMayor, columnaElementoMayor);

return 0;
}

```

Diagrama de flujo

