TP 3 LPOO PUNTEROS

Apellido y nombre: Ridolfi Ramiro

Ejercitación

1. Intercambiar el valor de dos variables usando punteros y funciones.

Mostrando los valores iniciales y valores finales.

#include <stdio.h>

void intercambio (int \*puntero1, int \*puntero2);

int main() {

int numero1 = 20, numero2 = 10;

printf("Valores iniciales\n numero 1 = %d \n numero 2 = %d\n", numero1, numero2);

intercambio(&numero1, &numero2);

printf("Valores finales\n numero 1 = %d \n numero 2 = %d", numero1, numero2);

}

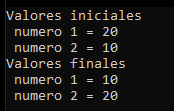
void intercambio (int \*puntero1, int \*puntero2) {

int temp = \*puntero1;

\*puntero1 = \*puntero2;

\*puntero2 = temp;

}



2. Encontrar la suma y el promedio de los elementos de un arreglo usando

punteros y funciones.

#include <stdio.h>

void encotrarSumaYPromedio(int \*array, int longitud, int \*suma, float \*promedio);

int main() {

int suma;

float promedio;

int array[] = {10, 20, 30, 40, 47};

int longitud = sizeof(array) / sizeof(array[0]);

encotrarSumaYPromedio(array, longitud, &suma, &promedio);

printf("La suma del arreglo es: %d\n", suma);

printf("El promedio del arreglo es: %f", promedio);

}

void encotrarSumaYPromedio(int \*array, int longitud, int \*suma, float \*promedio) {

int i;

\*suma = 0;

for (i = 0; i < longitud; i++) {

\*suma += \*(array + i);

}

\*promedio = (float)\*suma / longitud;

}



3. Encontrar el mayor y el menor elemento de un arreglo usando punteros y

funciones.

#include <stdio.h>

void encontrarMayorYMenor(int \*array, int longitud, int \*mayor, int \*menor);

int main() {

int mayor, menor;

int array[] = {10, 50, 100, 90, 20};

int longitud = sizeof(array) / sizeof(array[0]);

encontrarMayorYMenor(array, longitud, &mayor, &menor);

printf("El numero mas grande del array es: %d\n", mayor);

printf("El numero mas chico del array es: %d", menor);

}

void encontrarMayorYMenor(int \*array, int longitud, int \*mayor, int \*menor) {

int i;

\*mayor = \*menor = array[0];

for (i = 1; i < longitud; i++) {

if (array[i] > \*mayor) {

\*mayor = \*(array + i);

}

if (array[i] < \*menor) {

\*menor = \*(array + i);

}

}

}



4. Copiar un arreglo en otro arreglo usando punteros y funciones. Mostrando

el resultado final.

#include <stdio.h>

int i;

void copiarArray(int \*arr1, int \*arr2, int longitud);

int main() {

int arr1[] = {50, 30, 40, 120, 20, 22};

int longitud = sizeof(arr1) / sizeof(arr1[0]);

int arr2[longitud];

copiarArray(arr1, arr2, longitud);

printf("Array 1:\n\n");

for ( i = 0; i < longitud; i++) {

printf("Posicion %d: %d\t", i, arr1[i]);

}

printf("\n\nArray 2:\n\n");

for ( i = 0; i < longitud; i++) {

printf("Posicion %d: %d\t", i, arr2[i]);

}

}

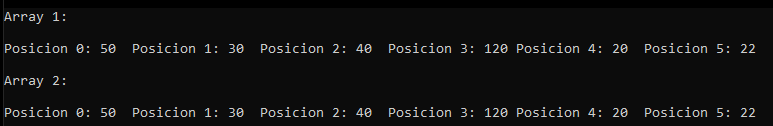
void copiarArray(int \*arr1, int \*arr2, int longitud) {

for ( i = 0; i < longitud; i++) {

\*(arr2 + i) = \*(arr1 + i);

}

}



5. Concatenar dos cadenas usando punteros y funciones.

#include <stdio.h>

#include <string.h>

void concatenar(char \*string1, char \*string2, char \*cadenaJunta);

int main() {

char string1[100] = "Hola, ";

char string2[100] = "buen dia!";

char cadenaJunta[200];

concatenar(string1, string2, cadenaJunta);

printf("String 1: %s\n", string1);

printf("String 2: %s\n", string2);

printf("\nEl string concatenado quedo asi: %s", cadenaJunta);

}

void concatenar(char \*string1, char \*string2, char \*cadenaJunta) {

int i, j;

for (i = 0; \*(string1 + i) != '\0'; i++) {

\*(cadenaJunta + i) = \*(string1 + i);

}

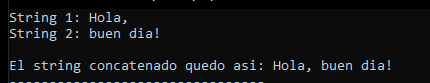
for (j = 0; \*(string2 + j) != '\0'; j++) {

\*(cadenaJunta + (i + j)) = \*(string2 + j);

}

\*(cadenaJunta + (i + j)) = '\0';

}



6. Contar el número de ocurrencias de un elemento en un arreglo usando

puntero y funciones

#include <stdio.h>

int i;

void contarOcurrencias(int \*arr, int longitud, int \*contador, int elemento);

int main() {

int contador = 0;

int arr[] = {20, 40, 20, 50, 20, 20, 100};

int longitud = sizeof(arr) / sizeof(arr[0]);

int elemento = 20;

contarOcurrencias(arr, longitud, &contador, elemento);

printf("Array completo.\n\n");

for (i = 0; i < longitud; i++) {

printf("Posicion %d: %d\t", i, arr[i]);

}

printf("\n\nCantidad de veces que se repite el elemento %d: %d", elemento, contador);

}

void contarOcurrencias(int \*arr, int longitud, int \*contador, int elemento) {

for (i = 0; i < longitud; i++) {

if (\*(arr + i) == elemento) {

(\*contador)++; // si fuera sin parentesis se estaria intentando aumentar la direccion de memoria.

}

}

}

