TP 2 LPOO CADENA DE CARACTERES

Apellido y nombre: Ridolfi Ramiro

**LINK DE GITHUB: https://github.com/Ridooo11/tp-2-enet35**

1. Pedir al usuario que ingrese su nombre y luego imprimirlo en pantalla.

#include <stdio.h>

#include <string.h>

int main() {

const int MAXIMO = 50;

char nombre[MAXIMO];

printf("Ingrese su nombre: ");

fgets(nombre, sizeof(nombre), stdin);

printf("Su nombre es: %s", nombre);

}



2. Realiza un programa que cuente cuántas vocales tiene una cadena de

caracteres ingresada por el usuario.

#include <stdio.h>

#include <string.h>

int main() {

char cadena[50];

const int CANT\_VOCALES = 5;

char vocales[5] = { 'a', 'e', 'i', 'o', 'u' };

printf("Ingrese una cadena de texto: ");

fgets(cadena, sizeof(cadena), stdin);

if (cadena[strlen(cadena) - 1] == '\n') {

cadena[strlen(cadena) - 1] = '\0';

}

int i, j;

int encontrado = 0;

for (i = 0; i < strlen(cadena); i++) {

for (j = 0; j < CANT\_VOCALES; j++) {

if (cadena[i] == vocales[j]) {

printf("Se encontro la vocal %c en la cadena\n", cadena[i]);

encontrado = 1;

break;

}

}

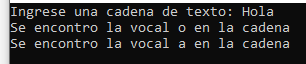
}

if (!encontrado) {

printf("No se ha encontrado ninguna vocal en la cadena: %s\n", cadena);

}

}





3. Realiza un programa que determine si una cadena de caracteres ingresada

por el usuario es palíndromo (se lee igual de izquierda a derecha que de

derecha a izquierda).

#include <stdio.h>

#include <string.h>

int main() {

char cadena[50];

char cadenaInvertida[50];

printf("Ingrese una cadena de texto: ");

fgets(cadena, sizeof(cadena), stdin);

int longitud = strlen(cadena);

int palindromo = 0;

if (cadena[longitud - 1] == '\n') {

cadena[longitud - 1] = '\0';

}

int i, j;

for (i = 0, j = longitud - 2; i < longitud; i++, j--) {

cadenaInvertida[j] = cadena[i];

}

cadenaInvertida[longitud] = '\0';

palindromo = strcmp(cadena, cadenaInvertida);

if (!palindromo) {

printf("La cadena de texto '%s' es palindroma", cadena);

} else {

printf("La cadena de texto '%s' no es palindroma", cadena);

}

}





4. Realiza un programa que reemplace todas las apariciones de un carácter

en una cadena de caracteres ingresada por el usuario por otro carácter

también ingresado por el usuario.

#include <stdio.h>

#include <string.h>

int main() {

char cadena[100];

char caracter, caracterNuevo;

int i;

int caracterExiste = 0;

printf("Ingrese una cadena de texto: ");

fgets(cadena, sizeof(cadena), stdin);

int longitud = strlen(cadena);

do {

printf("Ingrese el caracter que desea reemplazar: ");

scanf(" %c", &caracter);

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

if (cadena[i] == caracter) {

caracterExiste = 1;

break;

}

}

if (!caracterExiste) {

printf("\nEl caracter que desea reemplazar no existe en el string.\n\n");

}

} while(caracterExiste == 0);

if (caracterExiste) {

printf("Ingrese el nuevo caracter: ");

scanf(" %c", &caracterNuevo);

if (cadena[longitud - 1] == '\n') {

cadena[longitud - 1] = '\0';

}

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

if (cadena[i] == caracter) {

cadena[i] = caracterNuevo;

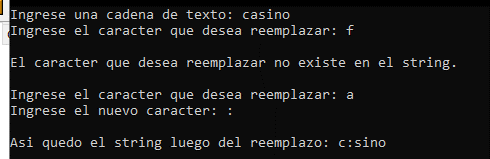
}

}

printf("\nAsi quedo el string luego del reemplazo: %s", cadena);

}

}



5. Realiza un programa que reemplace todas las apariciones de un carácter

en una cadena de caracteres ingresada por el usuario por otro carácter

también ingresado por el usuario.

6. Realiza un programa que determine si una cadena de caracteres ingresada

por el usuario es un anagrama de otra cadena también ingresada por el

usuario. Un anagrama es una palabra o frase formada por las mismas

letras de otra palabra o frase, pero en un orden diferente, como por

ejemplo roma y amor

#include <stdio.h>

#include <string.h>

int main() {

char cadena1[100], cadena2[100];

int contador[256] = {0};

int i;

printf("Ingresa la primera cadena: ");

scanf("%s", cadena1);

printf("Ingresa la segunda cadena: ");

scanf("%s", cadena2);

if (strlen(cadena1) != strlen(cadena2)) {

printf("Las cadenas no son anagramas\n");

} else {

for (i = 0; cadena1[i] && cadena2[i]; i++) {

contador[cadena1[i]]++;

contador[cadena2[i]]--;

}

for (i = 0; i < 256; i++) {

if (contador[i] != 0) {

printf("Las cadenas no son anagramas\n");

return 0;

}

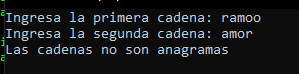
}

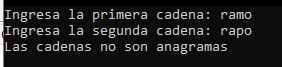
printf("Las cadenas son anagramas\n");

}

}







7. Realizar un programa que ingrese 5 palabras y devuelta la cadena mas

larga entre ellas

#include <stdio.h>

#include <string.h>

int main() {

char palabras[5][50];

char palabraMasLarga[50];

int i;

printf("Ingresa 5 palabras:\n");

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

printf("Palabra %d: ", i + 1);

scanf("%s", palabras[i]);

}

strcpy(palabraMasLarga, palabras[0]);

for (i = 1; i < 5; i++) {

if (strlen(palabras[i]) > strlen(palabraMasLarga)) {

strcpy(palabraMasLarga, palabras[i]);

}

}

printf("La palabra mas larga es: %s\n", palabraMasLarga);

}

