1)

char a[ ]=”hola”;

char b[ ]=”mundo”, c[10];

if (a==b)

printf(“son iguales. ”);

else

c=b;

No se puede por que no se puede compar string, Se debe usar strcmp(a,b).

DEBERIA SER

if (!strcmp(a,b)){

printf("son iguales. ");}

else{

strcpy(c,b);}

2)

char a[5]; LE ESTAS ASIGNANDO 5 ESPACIOS A EL VECTOR a;

char \*a; A LO QUE APUNTA a;

3). ¿Cuál es la diferencia entre estas tres declara

ciones?

char a[] = "hola"; CADENA

char a[4] = "hola"; ARREGLO

char a[] = {‘h’, ‘o’, ‘l’ ,’a’}; ARREGLO

char a[] = {‘h’, ‘o’, ‘l’ ,’a’, ‘\0’}; CADENA

char \*a = "hola"; APUNTA A UNA CADENA

4. Para cada una de las siguientes funciones descr

ibir los parámetros, que hacen y que archivo.h deb

e ser

incluido : strlen, strcmp, strcpy, strcat

STRLEN: DEVUELVE LA LONGITUD DE UNA CADENA.

**Declaración**: size\_t strlen(const char \*cadena)

**Parámetros**: La cadena a analizar

**Valor devuelto**: Un número entero

**Detalles**:

Devuelve la longitud (número de letras almacenadas) en una cadena de texto, sin contar el carácter nulo final

STRCMP:

**Declaración**: int strcmp(const char \*cadena1, const char \*cadena2);

**Parámetros**: Las dos cadenas a comparar

**Valor devuelto**: Un número entero

**Detalles**:

Devuelve 0 si las cadenas de texto son iguales (incluyendo mayúsculas y minúsculas); si la primera cadena es mayor que la segunda, devuelve un número positivo; si es mayor la segunda, devuelve un valor negativo. Existen variantes (no estándar, dependientes del compilador) que comparar dos cadenas despreciando mayúsculas y minúsculas, como stricmp

STRCAT:

**Declaración**: char \*strcat(char \*destino, const char \*origen);

**Parámetros**: La cadena inicial y la que se le quiere añadir

**Valor devuelto**: La cadena resultante

**Detalles**:

Devuelve el resultado de concatenar la segunda cadena a la primera (se puede ignorar: se modifica también la cadena que se pasa como parámetro)

STRCPY

**Declaración**: char \*strcpy(char \*destino, const char \*origen);

**Parámetros**: La cadena inicial y su nuevo valor (otra cadena)

**Valor devuelto**: La cadena resultante

**Detalles**:

Devuelve el nuevo valor de la cadena (se puede ignorar: se modifica también la cadena que se pasa como parámetro)