
 GOBIERNO DEL ESTADO DE MÉXICO	MANUAL DE PRÁCTICAS FO-TESJI-11100-12			 TECNOLÓGICO DE ESTUDIOS SUPERIORES <u>JILOTEPEC</u>	
Nombre de la práctica	CADENAS Y FUNCIONES			No.	12
Asignatura:	METODOS NUMERICOS	Carrera:	ING. SISTEMAS COMPUTACIONALES	Duración de la práctica (Hrs)	10

Nombre del alumno: Lissette Garcia Nolasco

I. Competencia(s) específica(s):

II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):

Aula

III. Material empleado:

Dev C++

Computadora

IV. Desarrollo de la práctica:

¿Cómo declarar una cadena?

-Crearla como un arreglo

Ejemplo:

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main() {
5      char cad[10];
6      int i;
7      printf ("INTRODUCE UNA PALABRA \n" );
8      gets (&cad);
9      printf ("---%s\n" , cad);
10
11     system ("Pause" );
12     return 0;
13 }
```

```

INTRODUCE UNA PALABRA
Yorely
---Yorely
Presione una tecla para continuar . . .
```

Crearla como arreglo sin tamaño

- char cad [] = "Es una cadena";

- ¿De cuantos elementos consta el arreglo cad? 14

Ejemplo 2:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main() {
5     char cad[]="Es una cadena" ;
6     printf ("%s\n", cad);
7     system("Pause");
8     return 0;
9 }
```

```
Es una cadena
Presione una tecla para continuar . . .
```

Ejemplo 3

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(){
5     int len;
6     char cad[]="ES UNA CADENA" ;
7     len=strlen(cad);
8     printf ("LA LONGITUD DE: \"%s\" ES: %d\n" , cad, len);
9     system("Pause");
10    return 0;
11 }
```

```
LA LONGITUD DE:|ES UNA CADENA|ES:13
Presione una tecla para continuar . . .
```

Ejercicio

- Escribe un programa que reciba una palabra por teclado.
- De acuerdo a la longitud de la palabra (N) que se ingreso por teclado imprime un cuadrado de asteriscos de (N x N).

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(){
5     char cad[]="";
6     int len,i,j;
7     puts(":3 INGRESE UNA PALABRA:");
8     gets(&cad);
9     len=strlen(cad);
10    printf("LA LONGITUD DE LA PALABRA ES %d\n",len);
11    for(i=0;i<len;i++){
12        for(j=0;j<len;j++){
13            printf("*");
14        }
15        puts("\n");
16    }
17    system("Pause");
18    return 0;
19 }
```

```
:3 INGRESE UNA PALABRA:
Yorely
LA LONGITUD DE LA PALABRA ES 6
*****

*****

*****

*****

*****

*****

Presione una tecla para continuar . . .
```

Funciones de cadena: strcpy

Copia el contenido de:

<cadena_origen> en <cadena_destino>.

strcpy(<cadena_destino>, <cadena_origen>)

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(){
5      int len;
6      char origen[]="Origen";
7      char destino[7];
8      strcpy(destino, origen);
9      printf("destino: %s ", destino);
10     system("Pause");
11     return 0;
12 }
```

```
destino:Origen
Presione una tecla para continuar . . .
```

Ejercicio

- Escribe un programa que reciba por teclado dos palabras.
- Y concatene N veces la segunda palabra a la primer palabra.
- Donde N es la longitud de la primera palabra.

Ej.

Palabra1: 'para'

Palabra2: 'brisas'

Palabra1: parabrisasbrisasbrisasbrisas

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(){
5     char antes []=" ";
6     char despues []=" ";
7     char guardar []=" ";
8
9     puts ("INGRESE UNA PALABRA:\n" );
10    scanf ("%s",&despues );
11    puts ("INGRESE OTRA PALABRA:\n" );
12    scanf ("%s",&antes );
13
14    strcpy ( guardar , despues );
15    strcpy ( despues , antes );
16
17    printf ( "\n1 Una palabra:%s\n1 Otra palabra:%s \n" , guardar , antes );
18    printf ( "\n2 Una palabra:%s\n2 Otra palabra:%s \n" , antes , guardar );
19
20    system ( "Pause" );
21    return 0;
22 }
```

```
INGRESE UNA PALABRA:
Yorely
INGRESE OTRA PALABRA:
Anna

1 Una palabra:Yorely
1 Otra palabra:Anna

2 Una palabra:Anna
2 Otra palabra:Yorely
Presione una tecla para continuar . . .
```

Funciones de cadena: strcat

Concatena el contenido de <cadena_origen> al final de <cadena_destino>

strcat(<cadena_destino>,<cadena_origen>)

Ejemplo 7

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(){
5      int len;
6      char origen []="BRISAS" ;
7      char destino [11]="PARA" ;
8      strcat (destino ,origen );
9      printf ("destino:%s\n " ,destino );
10     system ("Pause" );
11     return 0;
12 }
```

```
destino:PARABRISAS
Presione una tecla para continuar . . .
```

Ejercicio 8:

Escribe un programa que reciba por teclado dos palabras.

Y concatene N veces la segunda palabra a la primer palabra.

Donde N es la longitud de la primera palabra.

Ej.

Palabra1: 'para'

Palabra2: 'brisas'

Palabra1: parabrisasbrisasbrisasbrisas



GOBIERNO DEL
ESTADO DE MÉXICO

MANUAL DE PRÁCTICAS

FO-TESJI-11100-12



TECNOLÓGICO DE ESTUDIOS SUPERIORES
JILOTEPEC

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 int main(){
5     int len,i;
6
7     char uno[30]=" ";
8     char dos[30]=" ";
9
10    puts ("INGRESE UNA PALABRA:\n" );
11    scanf ("%s",&uno);
12    puts ("INGRESE OTRA PALABRA:\n" );
13    scanf ("%s",&dos);
14
15    len =strlen (uno);
16
17
18    printf ("\n1 Una palabra:%s\n", uno);
19    printf ("\n2 Otra palabra:%s\n", dos);
20    printf ("\n3 EL TAMAÑO DE LA PALABRA 1 ES:%d \n ", len);
21    for(i=0;i<len;i++){
22        strcat (uno,dos);
23    }
24    printf ("%s \n", uno);
25    system ("Pause");
26    return 0;
27 }
```

INGRESE UNA PALABRA:

para

INGRESE OTRA PALABRA:

brisas

1 Una palabra:para

Funciones de cadena: strcmp

Compara las dos cadenas y devuelve un 0 si las dos cadenas son iguales.

Un número positivo si <cadena1> es menor que <cadena2>

Un número negativo (mayor que cero) si <cadena1> es mayor que <cadena2>.

strcmp(<cadena1>, <cadena2>)

Ejemplo 9:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 int main(){
5     int res;
6
7     char str1[]="para" ;
8     char str2[]="brisas" ;
9
10    res =strcmp ( str1, str2);
11    printf ( "Resultado: %d\n" , res);
12
13    system ( "Pause" );
14    return 0;
15 }
```

Resultado:1
Presione una tecla para continuar . . .

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 int main(){
5     int res;
6
7     char str1[]="brisas" ;
8     char str2[]="para" ;
9
10    res =strcmp ( str1, str2);
11    printf ( "Resultado: %d\n" , res);
12
13    system ( "Pause" );
14    return 0;
15 }
```

Resultado:-1
Presione una tecla para continuar . . .

Ejemplo 11:

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  int main(){
5      int res;
6
7      char str1[]="para" ;
8      char str2[]="para" ;
9
10     res=strcmp(str1,str2);
11     printf("Resultado:%d\n",res);
12
13     system("Pause");
14     return 0;
15 }
```

```
Resultado:0
Presione una tecla para continuar . . .
```

Ejercicio



52

- ☐ Escribe un programa que reciba por teclado dos palabras y te indique cual de ellas es mayor y cual es la menor.
- ☐ En caso de ser iguales, que imprima la leyenda 'ambas palabras son iguales'.
- ☐ Compara las dos cadenas y devuelve un 0 si las dos cadenas son iguales.
- ☐ Un número negativo si <cadena1> es menor que <cadena2>
- ☐ Un número positivo (mayor que cero) si <cadena1> es mayor que <cadena2>.





```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 int main(){
5     int res;
6     char str1[]=" ";
7     char str2[]=" ";
8     puts ("INGRESE UNA PALABRA:\n" );
9     scanf ("%s",&str1);
10    puts ("INGRESE OTRA PALABRA:\n" );
11    scanf ("%s",&str2);
12    res =strcmp (str1,str2);
13
14    if(res>0||res<0){
15        printf ("Resultado:%d\n" ,res);
16    }
17    else{
18        printf ("AMBAS PALABRAS SON IGUALES\n" );
19        printf ("Resultado:%d\n" ,res);
20    }
21    if(str1>str2){
22        printf ("la primer palabra es mayor\n" ,res);
23    }
24    system ("Pause" );
25    return 0;
26 }
```

INGRESE UNA PALABRA:

perro

INGRESE OTRA PALABRA:

gato

Resultado:1

la primer palabra es mayor

Presione una tecla para continuar . . .

INGRESE UNA PALABRA:

gato

INGRESE OTRA PALABRA:

gato

AMBAS PALABRAS SON IGUALES

Resultado:0

la primer palabra es mayor

Presione una tecla para continuar . . .

Ejercicio



53

- Crea un programa que detecte una palabra palíndroma.
 - ▣ Los palíndromos son palabras que se leen igual de izquierda a derecha que de derecha a izquierda.
 - ▣ Ejemplo: ala, rotor, salas.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int main(){
4     int n, x, a=0, p=0, e=0, u;
5     char str1[20];
6     char str2[20];
7     char b=" ";
8     puts ("INGRESE UNA PALABRA: \n" );
9     gets (&str1);
10    n=strlen (str1);
11    for( u=0; u<=n; u++){
12        if( str1[u]==b){
13            u++;
14        }
15        str2[e]=str1[u];
16        e++;
17    }
18    x=0;
19    n=strlen (str2);
20    for( x=n-1; x>=0; x--){
21        if( str2[x]==str2[a]){
22            p++;
23        }
24        a++;
25    }
26    if( p==n){
27        printf ("ES UNA PALABRA POLINDROMA\n" );
28    }else{
29        printf ("NO ES UNA PALABRA POLINDROMA\n" );
30    }
31    system ("Pause" );
32    return 0;
33 }
34
```

INGRESE UNA PALABRA:

ala

ES UNA PALABRA POLINDROMA

Presione una tecla para continuar . . .

INGRESE UNA PALABRA:

gato

NO ES UNA PALABRA POLINDROMA

Presione una tecla para continuar . . .

Ejercicio

54

- Crea un programa que cuente cuantas ocurrencias de cada letra contiene una palabra.

■ Ejemplo:

Palabra

P → 1

a → 3

l → 1

b → 1

r → 1





```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 int main(){
5     char caracteres [150];
6     int palabra [200];
7     int i;
8     puts ("Ingrese una palabra" );
9     scanf ("%s", caracteres );
10    for (i=0; i<150; i++){
11        if ( caracteres [i]<0){
12            caracteres [i]=(i*-1);
13        }
14    }
15    for (i=0; i<200; i++){
16        palabra [i]=0;
17    }
18    for (i=0; caracteres [i]!='\0' ; i++){
19        palabra [caracteres [i]]++;
20    }
21    for (i=0 ; i<200; i++){
22        if ( palabra [i]>0){
23            printf ("%c = %i\n" , i, palabra [i]);
24        }
25    }
26    return 0;
27 }
```

Ingrese una palabra

Palabra

P = 1

a = 3

b = 1

l = 1

r = 1

Process exited after 6.108 seconds with return value 0
Presione una tecla para continuar . . .