

CADENAS

¿Qué es una cadena?

2

- Una **cadena** es un arreglo de caracteres. En donde, por lo general el último elemento deberá ser el carácter ‘\0’.

H	O	L	A		M	U	N	D	O
---	---	---	---	--	---	---	---	---	---

¿Cómo declarar una cadena?

3

□ Crearla como un arreglo

■ `char c [10];`


```
int main(){
    char cad[10];
    int i;
    printf("Introduce una palabra \n");
    scanf("%s", cad);
    printf("%s", cad);
    system("Pause");
    return 0;
}
```

¿Cómo declarar una cadena?

4

□ Crearla como un arreglo

■ `char c [10];`

```
int main(){  
     char cad[10];  
    printf("Introduce una palabra \n");  
    scanf("%s", cad);  
    printf("%s", cad);  
    system("Pause");  
    return 0;  
}
```

cad[0]	\0
cad[1]	04
cad[2]	\0
cad[3]	\0
cad[4]	01
cad[5]	\0
cad[6]	\0
cad[7]	01
cad[8]	\f
cad[9]	04

¿Cómo declarar una cadena?

□ Crearla como un arreglo

■ `char c [10];`

```
int main(){  
    char cad[10];  
    printf("Introduce una palabra \n");  
    scanf("%s", cad);  
    printf("%s", cad);  
    system("Pause");  
    return 0;  
}
```

Introduce una palabra

—

cad[0]	\0
cad[1]	04
cad[2]	\0
cad[3]	\0
cad[4]	01
cad[5]	\0
cad[6]	\0
cad[7]	01
cad[8]	\f
cad[9]	04

¿Cómo declarar una cadena?

6

□ Crearla como un arreglo

■ `char c [10];`

```
int main(){  
    char cad[10];  
    printf("Introduce una palabra \n");  
    scanf("%s", cad);  
    printf("\n %s", cad);  
    system("Pause");  
    return 0;  
}
```

Introduce una palabra
curso C

cad[0]	c
cad[1]	u
cad[2]	r
cad[3]	s
cad[4]	o
cad[5]	\0
cad[6]	\0
cad[7]	01
cad[8]	\f
cad[9]	04

¿Cómo declarar una cadena?

7

□ Crearla como un arreglo

■ `char c [10];`

```
int main(){  
    char cad[10];  
    printf("Introduce una palabra \n");  
    scanf("%s", cad);  
    printf("\n %s ", cad);  
    system("Pause");  
    return 0;  
}
```

Introduce una palabra
curso C
curso Presione cualquier tecla para continuar...


cad[0]	c
cad[1]	u
cad[2]	r
cad[3]	s
cad[4]	o
cad[5]	\0
cad[6]	\0
cad[7]	01
cad[8]	\f
cad[9]	04

¿Cómo declarar una cadena?

8

□ Crearla como un arreglo

■ `char c [10];`

```
int main(){  
     char cad[10];  
    printf("Introduce una palabra \n");  
    gets (cad);  
    printf("\n %s ", cad);  
    system("Pause");  
    return 0;  
}
```

cad[0]	\0
cad[1]	04
cad[2]	\0
cad[3]	\0
cad[4]	01
cad[5]	\0
cad[6]	\0
cad[7]	01
cad[8]	\f
cad[9]	04

¿Cómo declarar una cadena?

9

□ Crearla como un arreglo

■ `char c [10];`

```
int main(){  
    char cad[10];  
    printf("Introduce una palabra \n");  
    gets (cad);  
    printf("\n %s ", cad);  
    system("Pause");  
    return 0;  
}
```

Introduce una palabra


cad[0]	\0
cad[1]	04
cad[2]	\0
cad[3]	\0
cad[4]	01
cad[5]	\0
cad[6]	\0
cad[7]	01
cad[8]	\f
cad[9]	04

¿Cómo declarar una cadena?

10

□ Crearla como un arreglo

■ `char c [10];`

```
int main(){  
    char cad[10];  
    printf("Introduce una palabra \n");  
     gets (cad);  
    printf("\n %s ", cad);  
    system("Pause");  
    return 0;  
}
```

Introduce una palabra
curso C

cad[0]	c
cad[1]	u
cad[2]	r
cad[3]	s
cad[4]	o
cad[5]	
cad[6]	C
cad[7]	\0
cad[8]	\f
cad[9]	04

¿Cómo declarar una cadena?

11

□ Crearla como un arreglo

■ `char c [10];`

```
int main(){  
    char cad[10];  
    printf("Introduce una palabra \n");  
    gets (cad);  
    printf("\n %s ", cad);  
    system("Pause");  
    return 0;  
}
```

Introduce una palabra
curso C
curso C Presione cualquier tecla para continuar...

cad[0]	c
cad[1]	u
cad[2]	r
cad[3]	s
cad[4]	o
cad[5]	
cad[6]	C
cad[7]	\0
cad[8]	\f
cad[9]	04

¿Cómo declarar una cadena?

12

- Crearla como arreglo sin tamaño

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

- ¿De cuantos elementos consta el arreglo cad?



¿Cómo declarar una cadena?

13

□ Crearla como arreglo sin tamaño

■ `char cad [] = "Es una cadena";`

■ ¿De cuantos elementos consta el arreglo cad? 14

Cadena

14

```
int main(){  
    → char cad[]="Es una cadena";  
    printf("%s \n", cad);  
    system("Pause");  
    return 0;  
}
```

cad[0]	E
cad[1]	s
cad[2]	
cad[3]	u
cad[4]	n
cad[5]	a
cad[6]	
cad[7]	c
cad[8]	a
cad[9]	d
cad[10]	e
cad[11]	n
cad[12]	a
cad[13]	\0

Cadena

15

```
int main(){  
    char cad[]="Es una cadena";  
    printf("%s \n", cad);  
    system("Pause");  
    return 0;  
}
```

Es una cadena
Presione cualquier tecla para continuar...

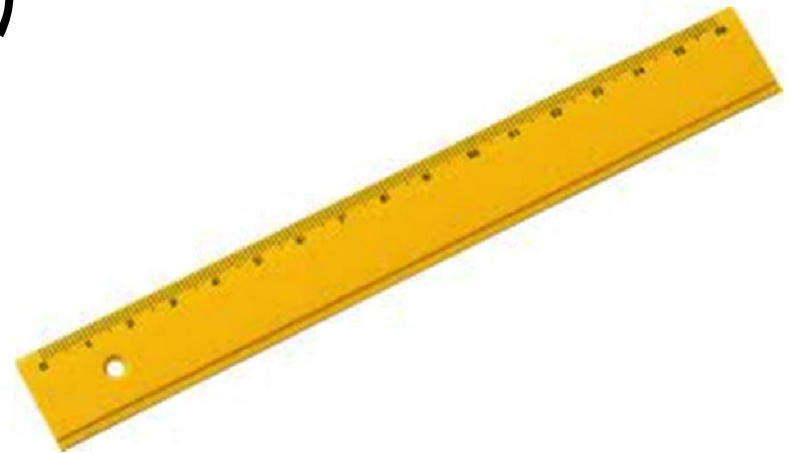
cad[0]	E
cad[1]	s
cad[2]	
cad[3]	u
cad[4]	n
cad[5]	a
cad[6]	
cad[7]	c
cad[8]	a
cad[9]	d
cad[10]	e
cad[11]	n
cad[12]	a
cad[13]	\0

Funciones de cadena: strlen

16

- Devuelve la longitud de la cadena sin tomar en cuenta el caracter de final de cadena.

`strlen(<cadena>)`



Ejemplo strlen

17

```
int main(){  
    → int len;  
    char cad[]="Es una cadena";  
    len = strlen(cad);  
    printf("La longitud de:\'%s\' es:  
    %d\n", cad, len);  
    system("Pause");  
    return 0;  
}
```

Es una cadena
Presione cualquier tecla para continuar...

len	

Ejemplo strlen

18

```
int main(){
    int len;
    → char cad[]="Es una cadena";
    len = strlen(cad);
    printf("La longitud de:\'%s\' es:
%d\n", cad, len);
    system("Pause");
    return 0;
}
```

Es una cadena
Presione cualquier tecla para continuar...

len	
cad[0]	E
cad[1]	s
cad[2]	
cad[3]	u
cad[4]	n
cad[5]	a
cad[6]	
cad[7]	c
cad[8]	a
cad[9]	d
cad[10]	e
cad[11]	n
cad[12]	a
cad[13]	\0

Ejemplo strlen

19

```
int main(){
    int len;
    char cad[]="Es una cadena";
    → len = strlen(cad);
    printf("La longitud de:\'%s\' es:
    %d\n", cad, len);
    system("Pause");
    return 0;
}
```

Es una cadena
Presione cualquier tecla para continuar...

len	13
cad[0]	E
cad[1]	s
cad[2]	
cad[3]	u
cad[4]	n
cad[5]	a
cad[6]	
cad[7]	c
cad[8]	a
cad[9]	d
cad[10]	e
cad[11]	n
cad[12]	a
cad[13]	\0

Ejemplo strlen

20

```
int main(){
    int len;
    char cad[]="Es una cadena";
    len = strlen(cad);
    printf("La longitud de:\'%s\' es:
%d\n", cad, len);
    system("Pause");
    return 0;
}
```

La longitud de: 'Es una cadena' es 13
Presione cualquier tecla para continuar...

len	
cad[0]	E
cad[1]	s
cad[2]	
cad[3]	u
cad[4]	n
cad[5]	a
cad[6]	
cad[7]	c
cad[8]	a
cad[9]	d
cad[10]	e
cad[11]	n
cad[12]	a
cad[13]	\0

Ejercicio

21

- 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 $(N \times N)$.

Funciones de cadena: strcpy

22

- Copia el contenido de:
 <cadena_origen> en <cadena_destino>.

`strcpy(<cadena_destino>, <cadena_origen>)`



Ejemplo strcpy

23

```
int main(){  
    → int len;  
    char origen[]="Origen";  
    char destino[7];  
    strcpy(destino, origen);  
    printf("destino: %s ", destino);  
    system("Pause");  
    return 0;  
}
```

len	

Ejemplo strcpy

24

```
int main(){
    int len;
    → char origen[]="Origen";
    char destino[7];
    strcpy(destino, origen);
    printf("destino: %s ", destino);
    system("Pause");
    return 0;
}
```

len	
origen[0]	O
origen[1]	r
origen[2]	i
origen[3]	g
origen[4]	e
origen[5]	n
origen[6]	\0

Ejemplo strcpy

25

```
int main(){
    int len;
    char origen[]="Origen";
    → char destino[7];
    strcpy(destino, origen);
    printf("destino: %s ", destino);
    system("Pause");
    return 0;
}
```

len	
origen[0]	O
origen[1]	r
origen[2]	i
origen[3]	g
origen[4]	e
origen[5]	n
origen[6]	\0
destino[0]	
destino[1]	
destino[2]	
destino[3]	
destino[4]	
destino[5]	
destino[6]	

Ejemplo strcpy

26

```
int main(){
    int len;
    char origen[]="Origen";
    char destino[7];
    → strcpy(destino, origen);
    printf("destino: %s ", destino);
    system("Pause");
    return 0;
}
```

len	
origen[0]	O
origen[1]	r
origen[2]	i
origen[3]	g
origen[4]	e
origen[5]	n
origen[6]	\0
destino[0]	O
destino[1]	r
destino[2]	i
destino[3]	g
destino[4]	e
destino[5]	n
destino[6]	\0

Ejemplo strcpy

27

```
int main(){
    int len;
    char origen[]="Origen";
    char destino[7];
    strcpy(destino, origen);
    printf("destino: %s ", destino);
    system("Pause");
    return 0;
}
```

destino: Origen
Presione cualquier tecla para continuar...

len	
origen[0]	O
origen[1]	r
origen[2]	i
origen[3]	g
origen[4]	e
origen[5]	n
origen[6]	\0
destino[0]	O
destino[1]	r
destino[2]	i
destino[3]	g
destino[4]	e
destino[5]	n
destino[6]	\0

Ejercicio

28

- Escribe un programa que reciba por teclado dos palabras y cada una de ellas las almacene en un arreglo.
- Después intercambia sus contenidos. Imprime el antes y el después
- Ej.

antes

Palabra_1 = ' Programación'

Palabra _2= 'Computadora'

después

Palabra_1 = 'Computadora'

Palabra _2= 'Programación'

Funciones de cadena: strcat

29

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

`strcat(<cadena_destino>, <cadena_origen>)`



Ejemplo strcpy


30

```
int main(){  
    → int len;  
    char origen[]="brisas";  
    char destino[11] = "para";  
    strcat(destino, origen);  
    printf("destino: %s ", destino);  
    system("Pause");  
    return 0;  
}
```

len	

Ejemplo strcpy

31

```
int main(){
    int len;
     char origen[]="brisas";
    char destino[11] = "para";
    strcat(destino, origen);
    printf("destino: %s ", destino);
    system("Pause");
    return 0;
}
```

len	
origen[0]	b
origen[1]	r
origen[2]	i
origen[3]	s
origen[4]	a
origen[5]	s
origen[6]	\0

Ejemplo strcpy

32

```
int main(){
    int len;
    char origen[]="brisas";
    → char destino[11] = "para";
    strcat(destino, origen);
    printf("destino: %s ", destino);
    system("Pause");
    return 0;
}
```

destino[8]	
destino[9]	
destino[10]	

len	
origen[0]	b
origen[1]	r
origen[2]	i
origen[3]	s
origen[4]	a
origen[5]	s
origen[6]	\0
destino[0]	p
destino[1]	a
destino[2]	r
destino[3]	a
destino[4]	\0
destino[5]	
destino[6]	
destino[7]	

Ejemplo strcpy

33

```
int main(){
    int len;
    char origen[]="brisas";
    char destino[11] = "para";
    → strcat(destino, origen);
    printf("destino: %s ", destino);
    system("Pause");
    return 0;
}
```

destino[8]	a
destino[9]	s
destino[10]	\0

len	
origen[0]	b
origen[1]	r
origen[2]	i
origen[3]	s
origen[4]	a
origen[5]	s
origen[6]	\0
destino[0]	p
destino[1]	a
destino[2]	r
destino[3]	a
destino[4]	b
destino[5]	r
destino[6]	i
destino[7]	s

Ejemplo strcat

34

```
int main(){
    int len;
    char origen[]="brisas";
    char destino[11] = "para";
    strcat(destino, origen);
    printf("destino: %s ", destino);
    system("Pause");
    return 0;
}
```

destino: parabrisas
Presione cualquier tecla para continuar...

destino[8]	a
destino[9]	s
destino[10]	\0

len	
origen[0]	b
origen[1]	r
origen[2]	i
origen[3]	s
origen[4]	a
origen[5]	s
origen[6]	\0
destino[0]	p
destino[1]	a
destino[2]	r
destino[3]	a
destino[4]	b
destino[5]	r
destino[6]	i
destino[7]	s

Ejercicio

35

- 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: para**brisas**brisas**brisas**brisas

Funciones de cadena: strcmp

36

- ❑ 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>.

`strcmp(<cadena1>, <cadena2>)`

Ejemplo strcmp

37

```
int main(){  
→ int res;  
  char str1[]="para";  
  char str2[] = "brisas";  
  res = strcmp( str1, str2 );  
  printf("resultado: %d ", res);  
  system("Pause");  
  return 0;  
}
```

res	

Ejemplo strcmp

38

```
int main(){
    int res;
    → char str1[]="para";
      char str2[] = "brisas";
      res = strcmp( str1, str2 );
      printf("resultado: %d ", res);
      system("Pause");
      return 0;
}
```

res	
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0

Ejemplo strcmp

39

```
int main(){
    int res;
    char str1[]="para";
    → char str2[] = "brisas";
    res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

res	
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0
str2[0]	b
str2[1]	r
str2[2]	i
str2[3]	s
str2[4]	a
str2[5]	s
str2[6]	\0

Ejemplo strcmp

str1 > str2

40

```
int main(){
    int res;
    char str1[]="para";
    char str2[] = "brisas";
    → res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

res	1
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0
str2[0]	b
str2[1]	r
str2[2]	i
str2[3]	s
str2[4]	a
str2[5]	s
str2[6]	\0

Ejemplo strcmp

str1 > str2

41

```
int main(){
    int res;
    char str1[]="para";
    char str2[] = "brisas";
    res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

resultado: 1
Presione cualquier tecla para continuar...

res	1
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0
str2[0]	b
str2[1]	r
str2[2]	i
str2[3]	s
str2[4]	a
str2[5]	s
str2[6]	\0

Ejemplo strcmp


42

```
int main(){  
→ int res;  
  char str1[]="brisas";  
  char str2[] = "para";  
  res = strcmp( str1, str2 );  
  printf("resultado: %d ", res);  
  system("Pause");  
  return 0;  
}
```

res	

Ejemplo strcmp


43

```
int main(){  
    int res;  
     char str1[]="brisas";  
    char str2[] = "para";  
    res = strcmp( str1, str2 );  
    printf("resultado: %d ", res);  
    system("Pause");  
    return 0;  
}
```

res	
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0

Ejemplo strcmp

44

```
int main(){
    int res;
    char str1[]="brisas";
     char str2[] = "para";
    res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

res	
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0
str2[0]	b
str2[1]	r
str2[2]	i
str2[3]	s
str2[4]	a
str2[5]	s
str2[6]	\0

Ejemplo strcmp

str1 < str2

45

```
int main(){
    int res;
    char str1[]="brisas";
    char str2[] = "para";
    → res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

res	-1
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0
str2[0]	b
str2[1]	r
str2[2]	i
str2[3]	s
str2[4]	a
str2[5]	s
str2[6]	\0

Ejemplo strcmp

str1 < str2

46

```
int main(){
    int res;
    char str1[]="brisas";
    char str2[] = "para";
    res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

resultado: -1
Presione cualquier tecla para continuar...

res	-1
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0
str2[0]	b
str2[1]	r
str2[2]	i
str2[3]	s
str2[4]	a
str2[5]	s
str2[6]	\0

Ejemplo strcmp


47

```
int main(){  
→ int res;  
  char str1[]="para";  
  char str2[] = "para";  
  res = strcmp( str1, str2 );  
  printf("resultado: %d ", res);  
  system("Pause");  
  return 0;  
}
```

res	

Ejemplo strcmp

48

```
int main(){
    int res;
     char str1[]="para";
    char str2[] = "para";
    res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

res	
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0

Ejemplo strcmp

49

```
int main(){
    int res;
    char str1[]="para";
    → char str2[] = "para";
    res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

res	
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0
str2[0]	p
str2[1]	a
str2[2]	r
str2[3]	a
str2[4]	\0

Ejemplo strcmp

str1 == str2

50

```
int main(){
    int res;
    char str1[]="brisas";
    char str2[] = "para";
    → res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

res	0
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0
str2[0]	p
str2[1]	a
str2[2]	r
str2[3]	a
str2[4]	\0

Ejemplo strcmp

str1 < str2

51

```
int main(){
    int res;
    char str1[]="brisas";
    char str2[] = "para";
    res = strcmp( str1, str2 );
    printf("resultado: %d ", res);
    system("Pause");
    return 0;
}
```

resultado: 0
Presione cualquier tecla para continuar...

res	0
str1[0]	p
str1[1]	a
str1[2]	r
str1[3]	a
str1[0]	\0
str2[0]	p
str2[1]	a
str2[2]	r
str2[3]	a
str2[4]	\0

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>`.



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.



08/11/2013

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

