Unidad 13

CADENAS

Recordatorio

- En la unidad anterior aprendimos acerca de arreglos para unir variables de un mismo tipo. Además vimos que una cadena es un arreglo donde el último elemento es el carácter nulo (\0)

Contenido

- Declaración de una cadena
- La longitud de una cadena
- Cómo copiar cadenas
- La lectura de cadenas con scanf()
- Las funciones gets() y puts()

Declaración de cadenas

- En esta sección estudiaremos como declarar e inicializar cadenas, así como la diferencia entre constantes de cadena y constantes de carácter.

Qué es una cadena

- En la unidad 12 vimos que una cadena es un arreglo que utiliza el carácter nulo (\0) al final del arreglo para indicar el fin de la cadena.
- Ejemplo arreglo de caracteres
- char A[3]={‘h’, ‘I’, ‘\0’}
- Esto es considerado como una cadena de caracteres por que tiene el carácter nulo.
- Donde \0 es un carácter de un byte

Constantes de cadenas

- Una cadena de caracteres encerradas entre comillas dobles se le denomina como constantes de cadenas y el compilador agrega automáticamente el carácter nulo.

- Ejemplo “hi” es una cadena de caracteres.

Inicialización de cadenas

- Podemos inicializar un arreglo de caracteres
- char A[2]={‘h’, ‘I’}

- Si se le agrega el carácter nulo entonces el arreglo es considerado como una cadena de caracteres.
- char A[3]={‘h’, ‘I’, ‘\0’}

Continuación de inicialización

- También se puede inicializar una arreglo de caracteres con una constante de cadena en lugar de una lista de constantes de carácter como en el ejemplo anterior

- Ejemplo
- char A[3]=“hi”
- El compilador agrega en forma automática el carácter nulo al final de hi. Además que tratará el arreglo de caracteres como una cadena.
- Note que se reserva el espacio para carácter nulo.

Otra opción 1

- Es dejar que el compilador automáticamente determine el tamaño de la cadena.
- char A[]=“hi”
- De esta manera, el compilador al inicializar determinara el espacio requerido en memoria agregando el carácter nulo al final de hi.

Opción 2

- También se puede un apuntador de tipo char y después asignarle una constante de cadena.
- Ejemplo
- *P=“Aprendiendo C”;
- ?Cuál es el tamaño de la cadena?

Importante

- No especificar el tamaño del arreglo de caracteres muy exacto, por que no se podrá agregar el carácter nulo.
- Ejemplo
- char cadena[4]=“nota”; // ilegal
- char cadena[5]=“nota”; // legal
- Además que puede que el compilador no marque advertencia pero al ejecutarlo produzca error.

Constantes de cadena vs constantes de carácter

- “Constantes de cadena” (serie de caracteres)
- ‘Constantes de carácter’ (un carácter)
- Inicialización
- char ch=‘x’;
- char cadena[]=“x”;

- Debido a que la cadena de caracteres es un arreglo se le puede asignar directamente una cadena a un apuntador (como vimos la clase anterior)
- char *p;
- p=“cadena”;
- Sin embargo
- p=‘c’; INCORRECTO
- Por que la variable p, espera un valor derecho e izquierdo. Y ‘c ‘ solo tiene valor derecho.

- Es válido asignar una constante de carácter a un apuntador tipo char indireccionado.

- char *p;
- *p=‘c’; CORRECTO

Longitud de una cadena

- Se puede determinar la longitud de una cadena usando la función strlen();
- Ver en la ayuda la sintaxis.
- Regresa el número de bytes sin tomar en cuenta el carácter nulo.

Cómo copiar cadenas

- Usando la función strcpy()
- Verificar la sintaxis en la ayuda
- Incluir librería string.h

- Escribir un programa que copie una cadena de un arreglo a otro
  - 1) haciendolo uno por uno
  - 2)usando la función strcopy

Lectura y escritura de cadenas

- Investiga la sintaxis y función de:

- gets()
- puts()
- Escribe un programa usando ambas

Lectura y escritura

- Investiga otras dos forma de leer e imprimir cadenas de caracteres (estas ya las han usado anteriormente)

- Escribe un programa usando las funciones encontradas.