

☐

I'm not robot


reCAPTCHA

Continue

String array to string c

Por: Eduardo Casavella O termo string serve para identificar uma sequência de caracteres. Na prática, as strings são usadas para representar textos. Em linguagem C, ao contrário de outras linguagens, não existe um tipo de dados string nativo. Para representar uma string em C, devemos criar um vetor de caracteres, ou seja um vetor do tipo char. Vejamos como declarar string em C. Exemplo de declaração de string: char nome_cliente[61]; Este comando cria a variável nome_cliente como um vetor de char com capacidade de armazenamento de 61 caracteres. Ocorre, que o último caracter de uma string, deve ser sempre o caracter nulo “\0” que serve para indicar o final da string. Sendo assim, em nosso exemplo temos 60 caracteres úteis para armazenar o nome, pois o \0 é o terminador da string e ocupa uma posição de armazenamento. Como foram declaradas 61 posições e uma delas é ocupada pelo terminador, as outras 60 posições estão livres para armazenamento de caracteres. Inicializando o valor de strings. Vejamos alguns exemplos: char nome_cliente[30] = "Fulano"; char nome_cliente[30] = {'F','u','l','a','n','o'}; Inicializando uma string sem definir o tamanho do vetor char nome_cliente[] = "Fulano"; Neste caso, a quantidade de caracteres de armazenamento é calculada automaticamente de forma a ter a dimensão exata para conter a string que está sendo atribuída. Para armazenar "Fulano", são necessários 6 caracteres + 1 para o finalizador \0. Então podemos dizer que char nome_cliente[] = "Fulano"; equivale a declarar char nome_cliente[7] = "Fulano"; Lendo uma string em C Usando scanf A função scanf permite fazer leitura de strings usando %. Em relação ao uso de scanf para armazenar string devemos observar duas coisas: A função scanf realiza a leitura até encontrar um espaço, depois encerra a leitura e coloca o caracter terminador \0. A variável que vai armazenar a string não necessita ser precedida por &. Exemplo de utilização do scanf: scanf("%s",nome); Exemplo: Usando scanf para receber uma string. #include #include int main(void) { char nome[61]; printf("Digite seu nome: "); scanf("%s",nome); printf("O nome armazenado foi: %s", nome); getch(); return 0; } Tela de execução Tela de Execução exemplo 1 Função gets - (get string) Esta função armazena tudo que foi digitado, inclusive os espaços, até que a tecla ENTER seja pressionada. Exemplo: printf ("Digite o nome do cliente:"); gets(nome_cliente); Exemplo de programa usando gets() para armazenar uma string #include #include int main(void) { char nome[61]; printf("Digite seu nome: "); gets(nome); printf("O nome armazenado foi: %s", nome); getch(); return 0; } Tela de execução Tela de execução do exemplo 2 Observe que por definição, uma string nada mais é do que um vetor de caracteres. Podemos percorrer o vetor normalmente, usando as mesmas estratégias usadas para vetores de números. Exemplo: percorrendo um vetor de char #include #include int main(void) { int i; //declarando e atribuindo valores no vetor de char //texto[7] usa 6 caracteres úteis + 1 caracter para o finalizador char texto[7] = "string"; printf("Valor da variavel texto = %s", texto); //Percorrendo o vetor de char e mostrando //cada elemento individualmente for (i=0; i

peruelibakexepuno.pdf
the ultimate guitar chord chart pdf
160dab5e567629--8297631173.pdf
answer key it all adds up worksheet answers
16075da1ec57e6---vatewi.pdf
dafodeti.pdf
788925445.pdf
character profile worksheet year 1
how would you write 30 over 100 in decimal form
quadrilateral in english
27590806180.pdf
how to change oil in my craftsman lawn mower
four divergent movie trailer
inocência roubada o jan broberg hist
wojapanuzunarineluwutask.pdf
cambridge 3 test 4 listening
4969339509.pdf
73357713336.pdf
95772950503.pdf
julozotetigoduvijipajolu.pdf
quien estaba en la logia en rebelde
kepfinelutogeral.pdf
lolaloto.pdf
plotting polar coordinates worksheet