Usuário: User

1 / 1

Data: Friday, October 6, 2023 at 5:02:01 PM - (Africa/Cairo)

When using strings, the computer creates an array of chars, containing the same number of cells as the length of the string + 1, and then fills the array with the ASCII codes of the letters of the string. In the last cell, the computer will store the ASCII code of the special character '\0

You're discussing how strings are implemented in the C programming language (and some other languages that borrow from C conventions).

In C, a string is represented as an array of characters (chars), and the string is terminated by a special character known as the null terminator, represented as '\0'. This null terminator is used to denote the end of the string, and it's crucial because C doesn't inherently know the length of arrays – it just knows where they start. By convention, when a C function encounters this '\0' character, it knows the string has ended.

To clarify:

1. A string like "hello" would be stored in memory as 'h', 'e', 'l', 'l', 'o', '\0'.
2. If you tried to determine the length of the string, it would return 5, even though there are 6 elements in the underlying array (the 5 characters of "hello" plus the null terminator).
3. The ASCII value of '\0' is 0. This is distinct from the ASCII value of the character '0', which is 48.

Always remember when working with C strings to account for this null terminator. If you're allocating memory for a string, for example, you need to allocate enough space for all the characters in the string, plus an additional space for the '\0' terminator.