

09. [Sebesta, 2000] Que exemplo pode ilustrar a falta de ortogonalidade no projeto do C?

Como exemplos da falta de ortogonalidade em uma linguagem de alto nível, manifestada como exceções à regra, consideremos as seguintes regras em C. Não obstante a linguagem C possua dois tipos de dados estruturados, arrays e registros (`structs`), registros podem ser retornados de funções, mas arrays não. Um membro de uma estrutura pode ser qualquer tipo de dado, exceto `void` ou uma estrutura do mesmo tipo. Um elemento de array pode ser qualquer tipo de dado, exceto `void` ou uma função. Parâmetros são passados por valor, a menos que sejam arrays, em cujo caso são, com efeito, passados por referência (porque o aparecimento do nome de um array isoladamente, ou seja, sem nenhum colchete, em um programa em C é interpretado como o endereço do primeiro elemento do array).

Uma expressão de adição simples, como $a + b$ usualmente significa que os valores de a e b são pegos da memória e adicionados. Porém, se acontecer de a ser um ponteiro, o valor pego de b pode ser modificado antes que a adição se desenvolva. Por exemplo, se a apontar para um valor que tenha dois bytes de extensão, o valor de b será multiplicado por dois antes que a adição desenvolva-se. O tipo de a , que é o contexto esquerdo de $a + b$ forçará o valor de b a ser modificado antes que ele seja adicionado a a .