

C预编译中关于字节对齐的问题Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022_C_E9_A2_84_E7_BC_96_E8_AF_91_c103_645395.htm 通过#pragma pack(n) 改变C编译器的字节对齐方式 在C语言中，结构是一种复合数据类型，其构成元素既可以是基本数据类型（如int、long、float等）的变量，也可以是一些复合数据类型（如数组、结构、联合等）的数据单元。在结构中，编译器为结构的每个成员按其自然对界（alignment）条件分配空间。各个成员按照它们被声明的顺序在内存中顺序存储，第一个成员的地址和整个结构的地址相同。例如，下面的结构各成员空间分配情况：struct test { char x1. short x2. float x3. char x4. }. 结构的第一个成员x1，其偏移地址为0，占据了第1个字节。第二个成员x2为short类型，其起始地址必须2字节对界，因此，编译器在x2和 x1之间填充了一个空字节。结构的第三个成员x3和第四个成员x4恰好落在其自然对界地址上，在它们前面不需要额外的填充字节。在test结构中，成员 x3要求4字节对界，是该结构所有成员中要求的最大对界单元，因而test结构的自然对界条件为4字节，编译器在成员x4后面填充了3个空字节。整个结构所占据空间为12字节。更改C编译器的缺省字节对齐方式 在缺省情况下，C编译器为每一个变量或是数据单元按其自然对界条件分配空间。一般地，可以通过下面的方法来改变缺省的对界条件：使用伪指令#pragma pack (n)，C编译器将按照n个字节对齐。使用伪指令#pragma pack ()，取消自定义字节对齐方式。另外，还有如下的一种方式：__attribute__((aligned (n)))，让所作用的结构成员对齐在n字节自

然边界上。如果结构中有成员的长度大于n，则按照最大成员的长度来对齐。 `__attribute__((packed))`，取消结构在编译过程中的优化对齐，按照实际占用字节数进行对齐。 以上的n = 1, 2, 4, 8, 16... 第一种方式较为常见。 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com