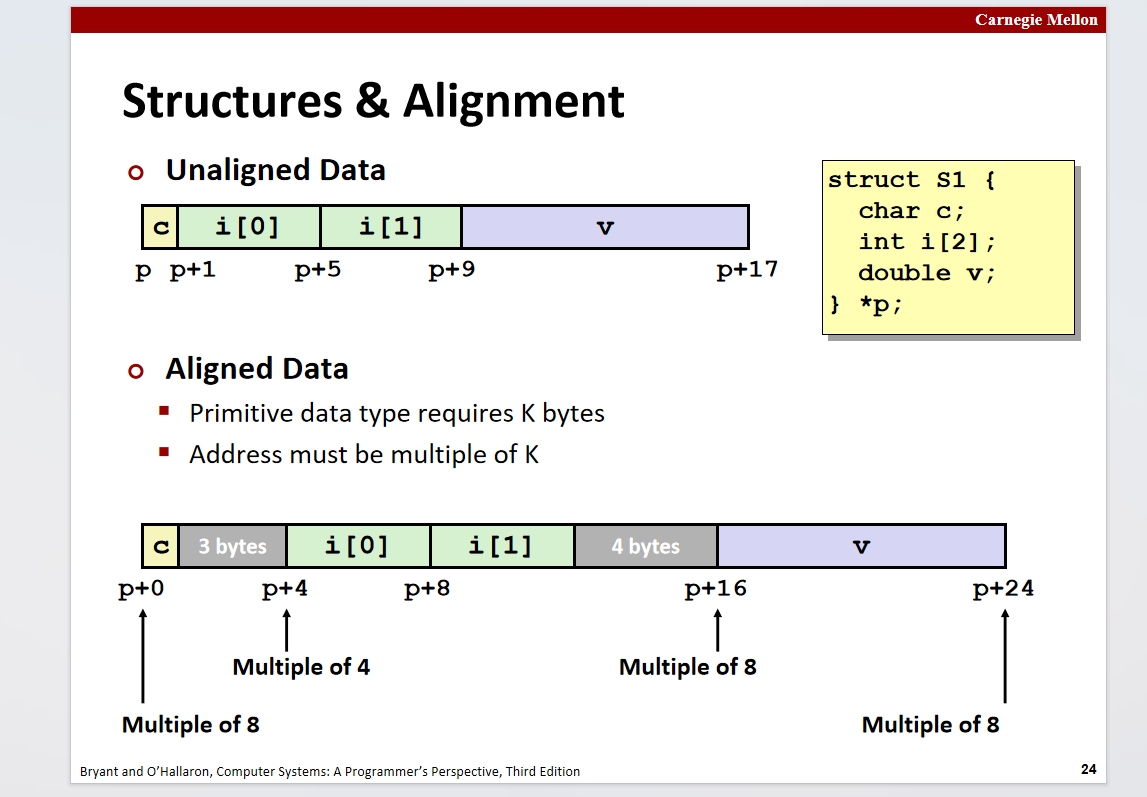
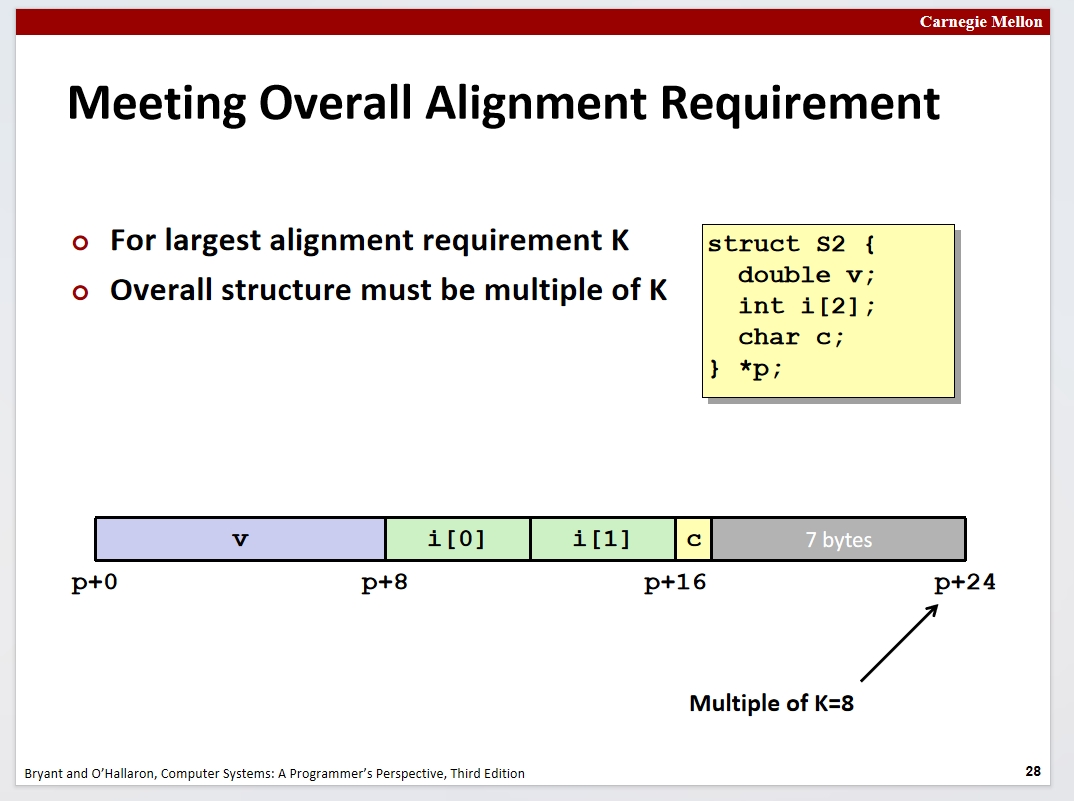
在结构体的初始化中，存在一种称为字节对齐的性质，即会将结构体的首地址放于一个最长长度的元素的整数倍的地址上，并且会将结尾的元素也放于这个倍数的地址上，基于这个原因，编译器会在创建时对其进行一个字节填充



如上图，不加以赘述

但可以看到，这样的话其实我们每一个结构体变量中都存在大量未被使用到的空间，我们可以通过一种技巧来优化，就是更改结构体中元素声明的顺序，如下图，这样能尽可能的减少结构体声明中会创建的空字节



对为什么这样做简单提一嘴，在程序中，一次读取的内存通常是一大块内存，就比如一次读取16个字的内存，而如果我们的结构体内存在这个过程中需要读俩次来实现，那么这会降低效率，且在一些系统上这个可能导致一些问题，因此会有这个规定，让结构体的地址与内存中的地址进行对齐