****

**计算机组成原理实验报告**

**题目：**数据的存放顺序和对齐方式

**姓 名： 臧祝利**

**专 业： 计算机科学与技术**

**年 级： 2020级**

**学 号： 202011998088**

**任课教师： 王志春**

**完成日期： 2022年05月13日**

**人工智能学院**

1. 实验要求

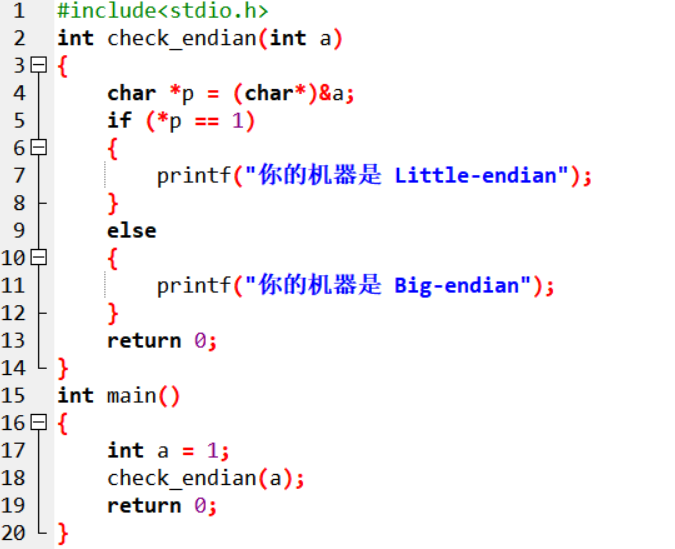
1.设计一个程序以检查你的机器是大端方式还是小端方式。

2.设计一个程序以检查内存变量（如结构或数组）是否按边界对齐。

1. 实验结果与分析

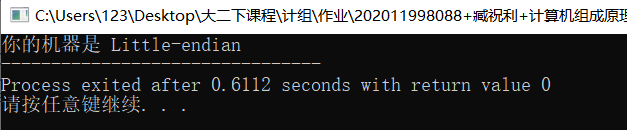
## 存放顺序

代码如图所示(Test.cpp)：



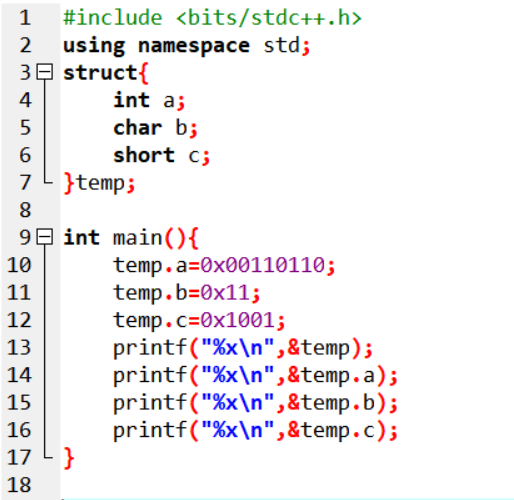
让指针p指向整数a的低地址，去看低地址的内容是什么，如果是1，说明低地址存放最低有效字节，那为小端方式，否则为大端方式。

根据程序的运行结果，可知为小端方式；

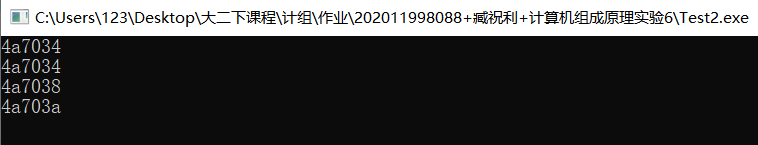


## 对齐方式

代码如图所示(Test2.cpp)：



运行结果如图所示，可知temp结构体里面，整型a的地址为4a7034，而a占据4个字节，因此字符b的地址为4a7038(4a7034+4)，但是字符b只占据1个字节，但是短整型c的地址却为4a703a(4a7038+2)，说明在b和c的地址之间有一个字节为空，说明数据的存储并不是根据存储空间顺序进行的，而是边界对齐的。



1. 实验小结

通过此次实验，可以得出以下结论：

我的机器采用的是小端方式，内存变量按边界对齐。

大端方式，是最高有效字节存放在低地址，由于平常描述一个数据时，默认左边为高位，因此大端方式很类似平常在纸面上对数据的描述——类似于顺序的一种形式；而小端方式，是最低有效字节存放在低地址。

在检测机器为小端方式还是大端方式时，一种方法是如上所示，使用一个字符型的指针强行指向数据的低地址，通过判断低地址的值来确定；也可以通过“联合体”的方式，在联合体内放一个char和int类型，由于二者共用存储空间，因此可以看一下字符里存储的是什么样的值，进而判断大小端；

边界对齐，是指在内存中，存储的变量并不是按照内存的顺序紧密靠在一起的，而地址往往是某个数（通常为4或8）的整数倍；通过上面的程序即可看出char和short之间存在一个空字节，因此为边界对齐。

1. 参考资料（如有参考资料可以列出）