

姓名	刘 宇
学号	2025211381

实验 成绩	
----------	--

华中师范大学计算学院 实 验 报 告 书

课程名称：高级语言程序设计

实验二

一、 问题描述（实验内容简述）

从键盘上输入一个长整型数，判断该长整型数的高 16 位所表示的有符号数及低 16 位所表示的有符号数的奇偶性。【用两种或以上方式实现，定义为函数，在 main 中调用】

二、实验环境（所使用的操作系统、编译器等）

操作系统：mac OS Sequoia 15.7.1

编译器：vs code

三、实现分析（实验原理分析，包括文字描述、流程图或伪代码等）

原理分析：

- 1.通过位运算符对数据重新进行排序和重组；
- 2.运算符的优先级；
- 3.printf 的格式化输出，scanf 的格式化输入；
- 4.在 main 函数中对自定义函数的调用；
- 5.判断奇偶性；
- 6.条件语句的使用。

四、实验代码（实验的核心代码）

```
#include<stdio.h>
void biteswap1(int a)
{
    int b;
    b = a>>16;
    if((b & 1) ==1)
    {
        printf("高十六位为奇数\n");
    }
}
```

```

    }
    else
        printf("高十六位为偶数\n");
}


void biteswap2(int a)
{
    int c;
    c = a & 0x0000ffff;
    if((c & 1) == 1)
    {
        printf("低十六位为奇数\n");
    }
    else
        printf("低十六位为偶数\n");
}

int main()
{
    int i;
    printf("input a HEX number:");
    scanf("%x",&i);
    biteswap1(i);
    biteswap2(i);
    return 0;
}

```

五、实验结果和分析（运行结果的截图，出现的错误、改正等分析）

运行成功截图：



```

supermilk@niurudeMacBook-Air 0 % cd "/Users/supermilk/0/" && gcc task2.c -o task2 && "/Users/supermilk/0/"task2
input a HEX number:0x12345675
高十六位为偶数
低十六位为奇数
supermilk@niurudeMacBook-Air 0 % cd "/Users/supermilk/0/" && gcc task2.c -o task2 && "/Users/supermilk/0/"task2
input a HEX number:0x12345678
高十六位为偶数
低十六位为偶数

```

运行出现的错误及错误代码截图：

```
task2.c:18:10: warning: & has lower precedence than ==; == will be evaluated first [-Wparentheses]
18 |     if(c & 1 ==1)
    |           ^~~~~
task2.c:18:10: note: place parentheses around the '==' expression to silence this warning
18 |     if(c & 1 ==1)
    |           ^
task2.c:18:10: note: place parentheses around the & expression to evaluate it first
18 |     if(c & 1 ==1)
    |           ^
    |           ( )
2 warnings generated.
```

```
#include<stdio.h>
void biteswap1(int a)
{
    int b;
    b = a<<16;
    if(b & 1 ==1)
    {
        printf("高十六位为偶数\n");
    }
    else
        printf("高十六位为奇数\n");
}
```

```
void biteswap2(int a)
{
    int c;
    c = a>>16 & 0x0000ffff;
    if(c & 1 ==1)
    {
        printf("低十六位为偶数\n");
    }
    else
        printf("低十六位为奇数\n");
}
```

错误原因：

1. 位运算符的优先级错误。

`c & 1 ==1` 此时会先算`==`级别更高，故 `c & 1` 要打括号。

2. 逻辑错误。

左图中，数据左移之后低位为 0，达不到取高 16 位的目的，且结果肯定始终为偶数，无法判断。

3. 输出值错误。

作“&”操作判断奇偶，结果为 1 应该为奇数，反之为偶数。

六、总结与改进

加强对概念的理解，熟悉运算符的优先级。