**作业：阅读以下3段C语言程序代码，运行结果，分析计算机中的定点数表示方法，在超星泛雅平台上提交PBL报告。**

**要求：**

**1）每道程序运行结果截屏；**

**2）结合已学知识点，对程序运行结果进行分析。**

**3）每人提交一份PBL报告，报告命名方式为：定点数表示方法报告-学号-姓名.doc/PDF**

**1.阅读并运行以下C语言程序源代码，打印结果，分析程序功能，讨论边界值、求相反数规则等问题。**

#include <stdio.h>

// 输出字符的十六进制编码

void char\_hex\_out(char a)

{

const char HEX[] = "0123456789ABCDEF";

int index = a & 0x0F;

printf("%c%c",HEX[(a&0xF0)>>4],HEX[a&0x0F]);

}

// 输出4个字节数据的十六进制编码，可用于输出4字节变量的机器码

void four\_byte\_out(char \* addr)

{

//输出指针变量的值，指针本质上是内存地址，是一个无符号数

//假设采用小端方式存储

char\_hex\_out(\* (addr+3));

char\_hex\_out(\* (addr+2));

char\_hex\_out(\* (addr+1));

char\_hex\_out(\* (addr+0));

printf("\n");

}

int main()

{

int a = -1;

int b = 2147483648;

int c = -b;

unsigned int d = -2147483648;

printf("a=%u=%d=0x%x \n",a,a,a);

printf("b=%u=%d=0x%x \n",b,b,b);

printf("c=%u=%d=0x%x \n",c,c,c);

printf("d=%u=%d=0x%x \n",d,d,d);

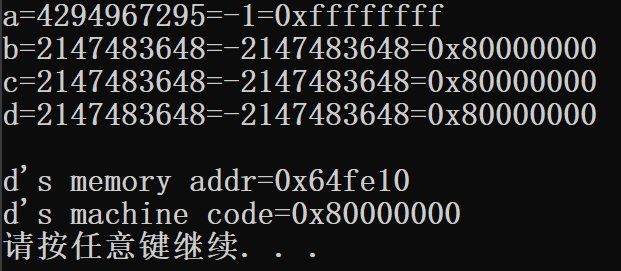
printf("\nd's memory addr=0x%x",&d); //变量d的虚存地址

printf("\nd's machine code=0x");

four\_byte\_out((char \*)(&d));         //输出变量d的机器码

return 0;

}



程序分别以无符号十进制整数，有符号十进制整数，十六进制整数输出a,b,c,d

1. **2.阅读并运行以下C语言程序源代码，打印结果，分析程序功能，讨论边界值问题。**

#include <stdio.h>

int main()

{

int a = 1;

int b = 2147483647;

int d = -1;

int e = -2147483648;

int c = a+b;

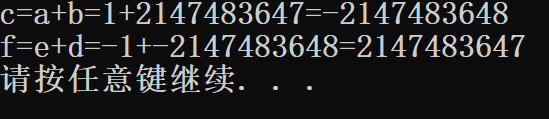
int f = e+d;

printf("c=a+b=%d+%d=%d \n",a,b,c);

printf("f=e+d=%d+%d=%d \n",d,e,f);

return 0;

}



1. **3.阅读并运行以下C语言程序源代码，打印结果，分析程序功能，讨论数据的内存分配问题。**

#include <stdio.h>

#include <stdlib.h>

int main()

{

int iarray[] = {5,4,3,2,1};

char \* iarr = (char \*) iarray;

printf("sizeof(char):%d\n",(int)sizeof(char));

printf("sizeof(short):%d\n",(int)sizeof(short));

printf("sizeof(int):%d\n",(int)sizeof(int));

printf("sizeof(unsigned int):%d\n",(int)sizeof(unsigned int));

printf("sizeof(long):%d\n",(int)sizeof(long));

printf("sizeof(long long):%d\n",(int)sizeof(long long));

printf("sizeof(size\_t):%d\n",(int)sizeof(size\_t));

printf("sizeof(void \*):%d\n",(int)sizeof(void \*));

printf("sizeof(iarray):%d\n",(int)sizeof(iarray));

printf("sizeof(iarr):%d\n",(int)sizeof(iarr));

return 0;

}

