Homework 1

- **Due: Nov. 5th**
- **Please submit your homework to e-learning server with format like 2430***.pdf**

1、Operations

(1)Now we have A: 0xF4, B: 0x11. Please compute A & B, A | B, A ^ B, ~A | ~B, A && B and A || B.

(2)Given two numbers x and y, I want to get a number that has the first half of x and the second half of y (such as x = 0x1111 1111, y = 0x0000 0000, result = 0x1111 0000). Please design a C program to achieve it.

(3)Shift operations.

Please fill in the following table.

x		x<<5		x>>3(logic)		x >>3(arithmetic)	
Hex	Binary	Binary	Hex	Binary	Hex	Binary	Hex
0xd1							
0x92							
0x4f							
0x36							

2、Align

Suppose the following code is executed on a **32-bit little-endian** machine, where “**int**” is 4 bytes, “**short**” is 2 bytes, “**char**” is 1 byte and “**pointer**” is 4 bytes.

(1) How many bytes are WASTED in struct s? Explain your solution.

```
struct s {
      char *name;
      short flags;
      union u {
         void *ptr;
            int a[2];
      } u;
      char c;
} s;
```

3、Assume we have following address binding table and value of registers

Address	**Value**	**Register**	**Value**
0xbffff0f8	0x00000001	%rax	0xc
0xbffff0fc	0xdeadbeef	%rbx	0xbffff108
0xbffff100	0x10	%rdx	0x4
0xbffff104	0x11	%rbp	0xbffff110
0xbffff108	0x12	%rsp	0xbffff100
0xbffff110	0xbffff138		
0xbffff114	0x8010240		
0xbffff120	0xbffff134		
0xbffff130	0x13		
0xbffff134	0x14		

0xbffff138	unknown		

Addressing

Please fill in the table below

Operand	Value
$0xbffff100	
0xbffff110	
%rbx	
(%rbx)	
(%rbx, %rax)	
0x4(%rsp, %rdx)	
-0x10(%rbp, %rdx, 4)	

Instructions

Suppose registers and bound values will be reset as above after each instruction. Please fill in the table below:

Instruction	Destination's Value
movq 0x4(%rbp, %rax), %rbx	%rbx =
movb %al, %bl	%rbx =
movw %bp, %bx	%rbx =
movsbq %bl, %rsp	%rsp =
movzbq %bl, %rsp	%rsp =
pushq %rbp	%rsp = (%rsp) =
popq %rax	%rsp = %rax= (%rsp) =

4、Assembly

Consider the following bit of C code and its part of disassembled IA64 machine code.

```
int main() {
1   char a[4] = "f";
2   char b[4];
3   int c = 2;
4   int d = someFunc(a, b, &c);
5   return 0;
}
```

```
someFunc:

pushq %rbp
movq %rsp,%rbp
movq %rdi,-0x8(%rbp)
movq %rsi,-0x10(%rbp)
movq %rdx,-0x18(%rbp)
movq -0x8(%rbp),%rax
movzbl (%rax),%edx
movq -0x10(%rbp),%rax
movb %dl,(%rax)
movq -0x18(%rbp),%rax
movq (%rax),%eax
leaq 0x1(%rax),%edx
movq -0x18(%rbp),%rax
movq %edx,(%rax)
movq $0x1,%eax
popq %rbp
retq
```

(1) Translate the assembly in the right column into C codes.

(2) Fill the table below when the C code executed in line 5

Variable	Variable’s value
b[0]	
c	
d	

Your C code:

5、y86

0x0104:	Func:
0x0104: a05f	pushl %ebp
0x0106: 2045	rrmovl %esp, %ebp
0x0108: a03f	__[1]__
0x010a: 30f3ffffffff	irmovl $-1, %ebx
0x0110: 501508000000	mrmovl 8(%ebp), %ecx
0x0116: 50250c000000	mrmovl 12(%ebp), %edx
0x011c: 6300	xorl %eax, %eax
0x011e: 6222	andl %edx, %edx
0x0120: 712e010000	jle End
0x0125:	Loop:
0x0125: __[2]__	addl %ecx, %eax
0x0127: 6032	addl %ebx, __[3]__
0x0129: __[4]__	jne Loop
0x012e:	End:
0x012e: b03f	popl %ebx
0x0130: 2054	rrmovl %ebp, %esp
0x0132: b05f	popl %ebp
0x0134: __[5]__	ret
0x0135:	Main:
0x0135: f0	brk
0x0136: 30f002000000	irmovl $2, %eax
0x013c: 30f305000000	irmovl __[6]__, %ebx
0x0142: a03f	pushl %ebx
0x0144: a00f	pushl %eax
0x0146: __[7]__	call Func
0x014b: 2054	rrmovl %ebp, %esp

(1) Please fill in the blanks within above Y86 binary and assembly code.

[1] ________________ [2] ________________ [3] ________________

[4] ________________ [5] ________________ [6] ________________

[7] ________________

(2) Please describe the function or purpose of Func,and provide the equivalent C code.