**<C언어로 ALU 함수 구현하기>**

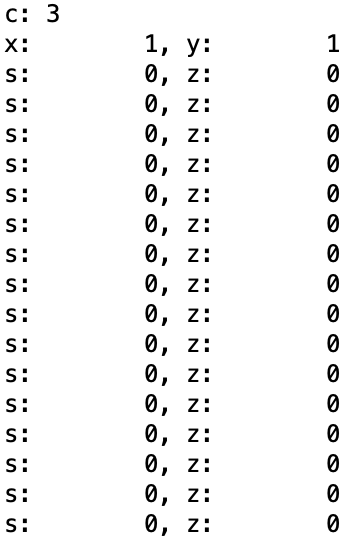
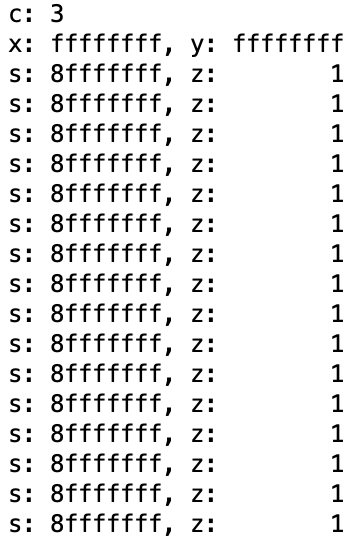
**201703091, 컴퓨터.전자시스템공학부, 전기범**

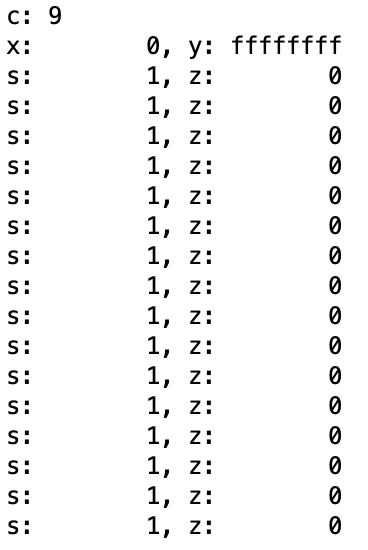
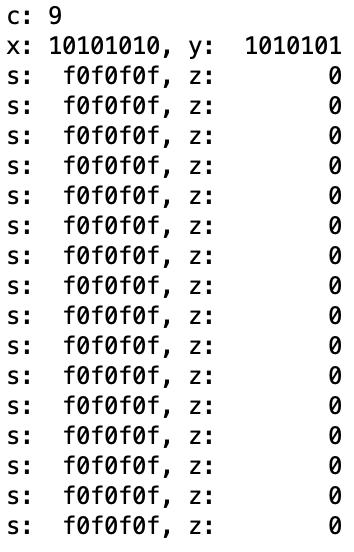
**제출일 : 2019/04/09**

**1. 소스코드**

#include **<stdio.h>**#include **<stdlib.h>  
  
int** logicOperation(**int** X, **int** Y, **int** C)  
{  
 **if** (C < 0 || C > 3) {  
 printf(**"error in logic operation\n"**);  
 exit(1);  
 }  
 **if** (C == 0) *// AND* **return** X & Y;  
 **else if** (C == 1) *// OR* **return** X | Y;  
 **else if** (C == 2) *// XOR* **return** X ^ Y;  
 **else** *// NOR* **return** ~(X | Y);  
}  
  
**int** addSubtract(**int** X, **int** Y, **int** C)  
{  
 **int** ret;  
 **if** (C<0||C>1){  
 printf(**"error in add/subtract operation\n"**);  
 exit(1);  
 }  
 **if** (C==0){  
 ret = X+Y;  
 }  
 **else**{  
 ret = X-Y;  
 }  
 **return** ret;  
}  
  
**int** shiftOperation (**int** V, **int** Y, **int** C)  
{  
 **int** ret;  
 **if** (C < 0 || C > 3) {  
 printf(**"error in shift operation\n"**);  
 exit(1);  
 }  
 **if** (C == 0) *// No shift* ret = V;  
 **else if** (C == 1) *// Logical Left* ret = V << Y;  
 **else if** (C == 2) *// Logical Right* ret = V >> Y;  
 **else** { *// Arithmetic Right* **if** (V>>Y < 0)  
 ret = (V>>Y) + 0x90000000;  
 **else** ret = V >> Y;  
 }  
 **return** ret;  
}  
  
**int** setLess(**int** X, **int** Y)  
{  
 **if**(X<Y) **return** 1;  
 **else return** 0;  
}  
  
**int** ALU(**int** X, **int** Y, **int** C, **int** \*Z)  
{  
 **int** c32, c10;  
 **int** ret;  
  
 c32 = (C >> 2) & 3;  
 c10 = C & 3;  
 **if** (c32 == 0) { *// shift* ret = shiftOperation(X, Y, c10);  
 } **else if** (c32 == 1) { *// set less* ret = setLess(X,Y);  
 } **else if** (c32 == 2) { *// addsubtract* **if** (addSubtract(X,Y,c10)==0) {  
 \*Z = 0;  
 ret = addSubtract(X, Y, c10);  
 }  
 **else** {  
 \*Z = 1;  
 ret = addSubtract(X, Y, c10);  
 }  
 } **else** { *// logic* ret = logicOperation(X, Y, c10);  
 }  
 **if** (ret>=0)  
 \*Z=0;  
 **else** \*Z=1;  
 **return** ret;  
}  
  
**void** test(**void**)  
{  
 **int** x,y,c,s,z;  
  
 x=0x00000000;  
 y=0xffffffff;  
 c=9;  
  
 printf(**"c: %d\n"**,c);  
 printf(**"x: %8x, y: %8x\n"**,x,y);  
 **for** (**int** i=0;i<16;i++){  
 s=ALU(x,y,c,&z);  
 printf(**"s: %8x, z: %8x\n"**,s,z);  
 }  
}  
  
**int** main(**void**)  
{  
 test();  
 **return** 0;  
}

**2. 실행결과**

** **

** **