HW0505

```c
#include <stdint.h>
#include <stdio.h>

// Assumes little endian
void printBits(size_t const size, void const * const ptr)
{
    unsigned char *b = (unsigned char*) ptr;
    unsigned char byte;
    int i, j;

    for (i = size-1; i >= 0; i--) {
        for (j = 7; j >= 0; j--) {
            byte = (b[i] >> j) & 1;
            printf("%u", byte);
        }
    }
    puts("");
}

unsigned int ui = 0;
unsigned short us = 0;
signed int si = -1;

int main()
{
    printf(  "ui in unsigned int   = "); printBits(sizeof(ui), &ui);
    printf("\nus in unsigned short = "); printBits(sizeof(us), &us);
    printf("\nsi in  signed  int   = "); printBits(sizeof(si), &si);
    printf("\n\n");

    int64_t r1 = ui + si;
    int64_t r2 = us + si;

    printf("%ld %ld\n", r1, r2);
    printf("\n");
    printf("r1 in binary = "); printBits(sizeof(r1), &r1);
    printf("r2 in binary = "); printBits(sizeof(r2), &r2);
    printf("\n\n");
    ////////////////////////////////////////////////

    printf("What if we convert them into int64 form : \n");
    r1 = ui;
    printf("ui in int64 form = "); printBits(sizeof(r1), &r1);
    r1 = us;
    printf("us in int64 form = "); printBits(sizeof(r1), &r1);
    r1 = si;
    printf("si in int64 form = "); printBits(sizeof(r1), &r1);

    return 0;
}
```

```
beck@DESKTOP-TQ4LO5K:/mnt/c/Users/Beck/OneDrive/桌面/2021-1/40947011S_HW05$ gcc -o test test.c
beck@DESKTOP-TQ4LO5K:/mnt/c/Users/Beck/OneDrive/桌面/2021-1/40947011S_HW05$ ./test
ui in unsigned int   = 00000000000000000000000000000000

us in unsigned short = 0000000000000000

si in  signed  int   = 11111111111111111111111111111111


4294967295 -1

r1 in binary = 00000000000000000000000000000000111111111111111111111111111111111
r2 in binary = 11111111111111111111111111111111111111111111111111111111111111111


What if we convert them into int64 form :
ui in int64 form = 0000000000000000000000000000000000000000000000000000000000000000
us in int64 form = 0000000000000000000000000000000000000000000000000000000000000000
si in int64 form = 1111111111111111111111111111111111111111111111111111111111111111
beck@DESKTOP-TQ4LO5K:/mnt/c/Users/Beck/OneDrive/桌面/2021-1/40947011S_HW05$
```

由上例可知，

在兩 integer 相加時，若擴展記憶體時，高位位元將保持 0

在 short 和 integer 相加時，若擴展記憶體時，高位位元將補 1