Homework 1

T1

1. Convert these decimal numbers to 8-bit 2's complement numbers:
将这些十进制数转换为 8 位二进制的二进制补码（2’ s complement）形式。
-103
+76

2. Convert the following 8-bit 2's complement numbers to decimal.
将下面的 8 位二进制二进制补码数转换为十进制数。
00110110
11101101

T2

1. What's the smallest and largest number that can be represented by an 8-bit 2's complement number? (Answer in decimal)
用 8 位二进制的二进制补码表示时,能表示的最小值和最大值分别是多少?（答案用十进制表示）

2. Try to determine the range that an N-bit 2's complement number can represent. (Answer in decimal)
试着确定一个 N 位二进制补码数所能表示的数值范围。（答案用十进制表示）

T3

Is there a negative integer that has identical 2's complement representation and original code in binary (8-bit)? If so, what is it? (Answer in decimal)
在 8 位二进制中，是否存在一个负整数，它的 二进制补码表示与 原码（sign-magnitude code） 表示相同？如果有，请写出这个数的十进制

T4

The C code below takes two integers and prints out whether their sum is negative or not.(The numbers given are guaranteed to be in the range of int.)
下面的 C 代码读取两个整数并判断它们的和是否为负数。(输入的数保证在 int 范围内。)

```c
#include <stdio.h>

int main(void) {
    int a, b;
    scanf("%d %d", &a, &b);
    if (a + b < 0)
        printf("Sum is negative\n");
    else
        printf("Sum is non-negative\n");
    return 0;
}
```

1. Under what circumstances will the program print "Sum is negative" while the actual mathematical sum a + b is positive?
在什么情况下，程序会输出 "Sum is negative"，但实际上 a + b 的数学结果是正数？（提示：考虑溢出 overflow 情况）

2. What happens if we change both variables to unsigned int instead of int?
如果把 a 和 b 都改为 unsigned int，程序输出会有什么变化？

T5

Write the decimal equivalents for the IEEE floating point number below.
为下面的 IEEE 浮点数写出其十进制等价形式。
0 10000010 10100000000000000000000

T6

What is the smallest number that can be represented in IEEE floating point format with 32 bits regardless of infinity? What about the smallest positive number? (Answer in binary)
在不考虑无穷大的情况下，IEEE 32 位浮点数格式中可以表示的最小的数是什么？那最小的正数又是什么？（请用二进制回答）

T7

Can you list all the integers whose IEEE floating point representations are exactly the same as their 2's complement integer representations? (Answer in decimal)
请列出所有这样的整数：它们的 IEEE 浮点数表示与它们的二进制补码整数表示**完全相同**。（答案用十进制表示）

T8

The following code uses three addition and subtraction operations to swap two integers.
下面的代码使用三次加减法操作来交换两个整数。

```
1 void swap(int *a, int *b) {
2     *a = *a + *b;
3     *b = _________;
4     *a = _________;
5 }
```

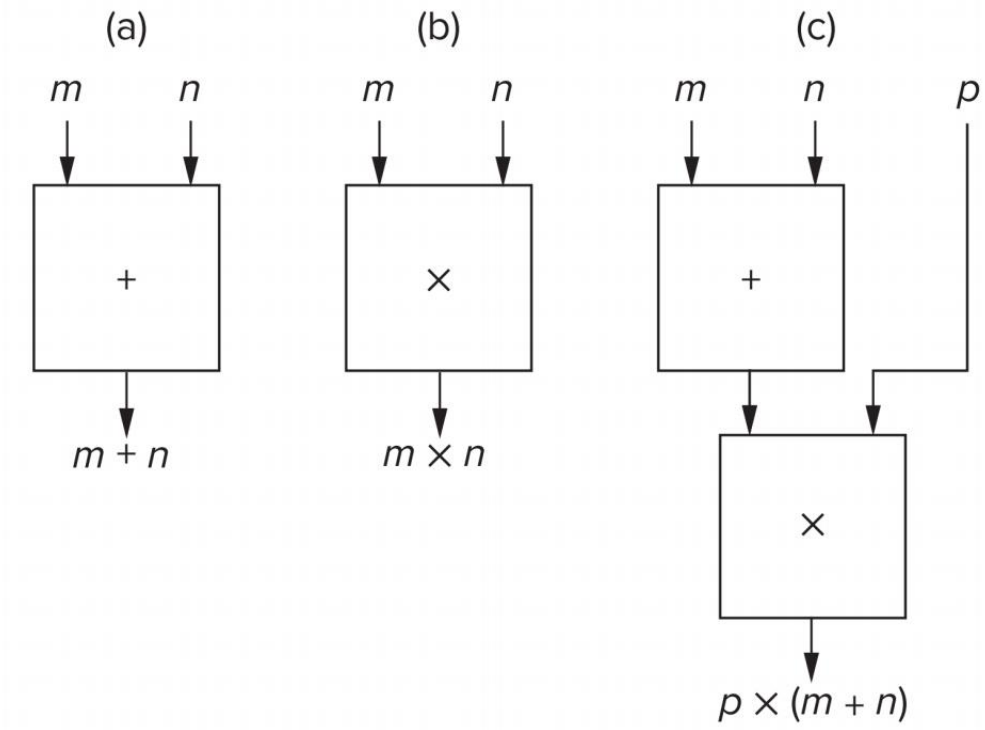
1. Fill in the blanks to complete the swap function.
填空以完成 swap 函数。

2. If this swap function is used in a sort function, is there any potential problem? If so, explain why and how to fix it.
如果在 sort 函数中使用这个 swap，是否存在潜在问题？请说明原因并给出修正方法

T9

We've got 2 blackboxes, each of which takes two numbers as input and produces a number as the output. The first one is capable of adding, while the second one is capable of multiplying. We can combine these blackboxes to calculate (m + n) × p.

我们有两个黑盒子，每个黑盒子都接收两个数字作为输入，并输出一个数字。第一个黑盒子可以执行**加法**运算，而第二个黑盒子可以执行**乘法**运算。
我们可以将这两个黑盒子组合起来，计算表达式 **(m + n) × p**。



You're required to draw circuits that can calculate the following expressions:

请你画出能够计算以下表达式的电路图：

- y=(a+b)×(c+d)
- z=(m+n+p)×q
- E = (x + y)^2 = (x + y) × (x + y)
- S=(a+b+c)×(d+e)

T10

We'd like to use binary to represent the following characters:
A to Z
a to z
0 to 9
2 special characters: (space) and .
1. How many bits do we need to represent a single character?
2. How many bits do we need to represent a string of characters?
3. Assume that we use 0 to represent A , 1 to represent B , and so on. So we use 63 to represent . .
What is the binary representation of “Ics 2025.” ?

我们想用二进制来表示以下字符：

A 到 Z
a 到 z
0 到 9
两个特殊字符：空格（space）和句号（.）

- 表示一个单独字符需要多少位（bits）？
- 表示一个字符串需要多少位？
- 假设我们用 0 表示 A，1 表示 B，以此类推。因此我们用 63 来表示句号（.）。

“Ics 2025.” 的二进制表示是什么？