# Recitation 5 Code

## From Xiao Liu

### Exercise: Array of Function Pointers

- An array is a data structure that stores a collection of identical data types. Similar to other data types, we can **create an array to store function pointers**.
- Given two operands of type `int` (you need to read them using `scanf`), we need to perform four operations on them sequentially: addition, subtraction, multiplication, and division. Design a structure that utilizes function pointers for these operations.

```c
#include "stdio.h"

double add(int, int);
double multiply(int,int);
double divide(int,int);
double subtract(int,int);

int main(void){
    printf("Please input two integers:");
    int a, b;
    scanf("%d%d", &a, &b);

    double (*operation[4]) (int, int);
    operation[0] = add;
    operation[1] = subtract;
    operation[2] = multiply;
    operation[3] = divide;

    for(int i = 0; i != 4; ++i){
        printf("The result of operation%d is: %lf.\n", i, (*operation[i])(a, b));
    }
}

double add(int a, int b) {
    return (double)a + (double)b;
}

double subtract(int a, int b) {
    return (double)a - (double)b;
}

double multiply(int a, int b) {
    return (double)a * b;
}

double divide(int a, int b) {
    return (double)a / (double)b;
```

```
}
```

You can even determine the function you used at runtime according to your input.

```
#include "stdio.h"

double add(int, int);
double multiply(int,int);
double divide(int,int);
double subtract(int,int);

int main(void){
    printf("Please input two integers:");
    int a, b;
    scanf("%d%d", &a, &b);

    double (*operation[4]) (int, int);
    operation[0] = add;
    operation[1] = subtract;
    operation[2] = multiply;
    operation[3] = divide;

    printf("Please input the operation number(0-3):\n");
    fflush(stdin);  // emptyt the buffer
    int operation_num;
    scanf("%d", &operation_num);
    printf("The result of operation%d is: %lf.\n", operation_num,
(*operation[operation_num])(a, b));
}

double add(int a, int b) {
    return (double)a + (double)b;
}

double subtract(int a, int b) {
    return (double)a - (double)b;
}

double multiply(int a, int b) {
    return (double)a * b;
}

double divide(int a, int b) {
    return (double)a / (double)b;
}
```