

**Національний технічний університет України “Київський  
політехнічний інститут”**

**Факультет прикладної математики  
Кафедра системного програмування і спеціалізованих  
комп’ютерних систем**

**ЛАБОРАТОРНА РОБОТА №2.3**

**з дисципліни  
“Структури даних і алгоритми”**

**ТЕМА: “МОДУЛІ”**

**Група: КВ-34  
Виконав :  
Фесенко Денис**

**Київ – 2024**

**1.Згідно заданої схеми взаємозв'язків модулів (Mi, MErr) і функцій (Fi, Err) намалювати повну схему взаємозв'язків модулів і функцій, на якій різними типами стрілок позначити напрями імпортування модулів та виклики функцій.**

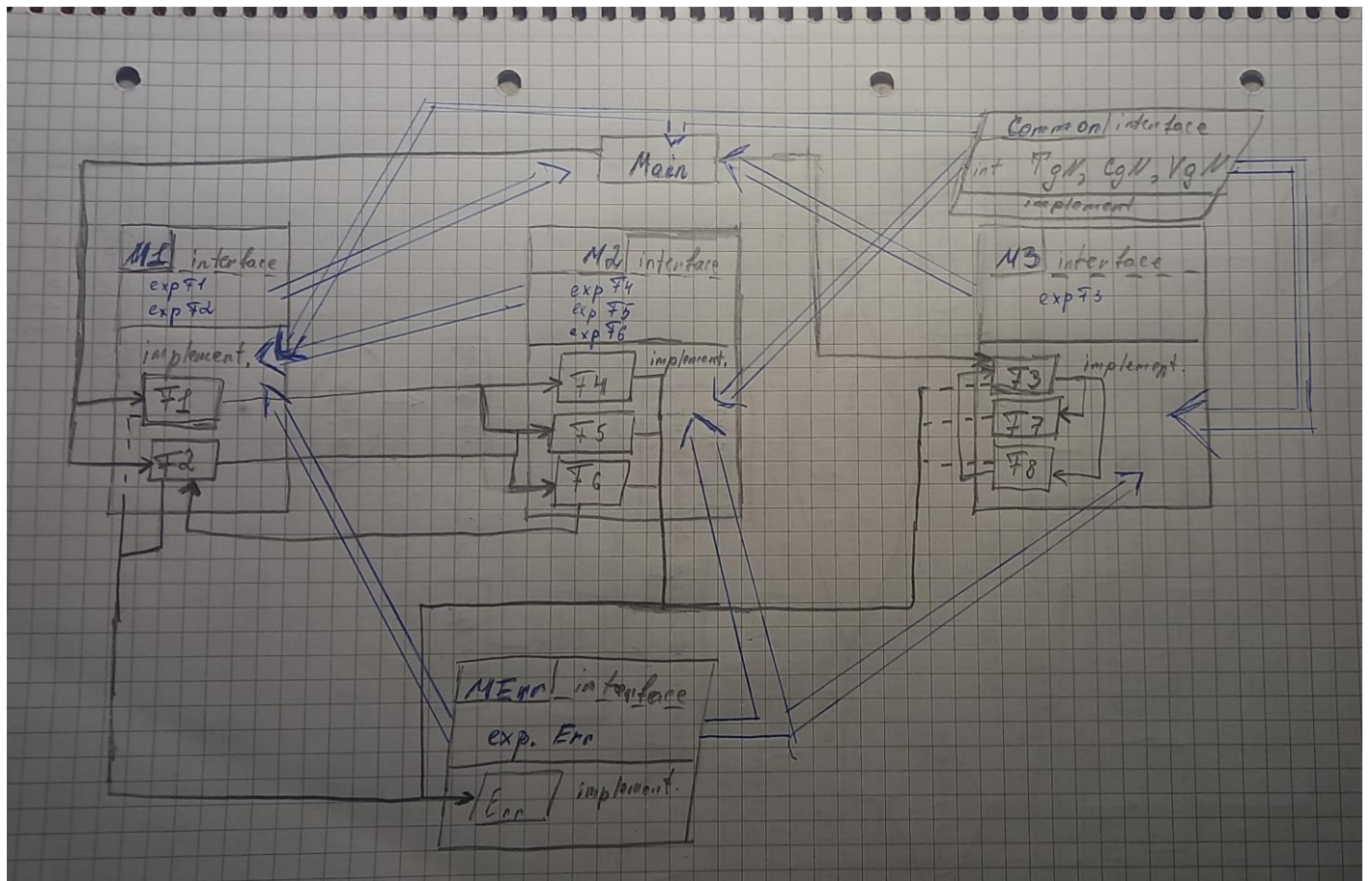
### 3. Функція обробки помилок Err повинна викликатися з усіх інших функцій.

**5.Реалізувати взаєморекурсивний виклик функцій, заданий на схемі, не менше 2 разів кожної із взаєморекурсивних функцій. Для цього реалізувати лічильник взаєморекурсивних викликів.**

## 7.Протокол повідомлень про початок та закінчення роботи функцій вивести у текстовий файл.

## 2

## Повна схема взаємозв'язків



## Код програми:

### common.h:

```
#ifndef __COMMON_H__
#define __COMMON_H__
#include <stdio.h>
#include <stdlib.h>

typedef int Tg1;
typedef int Tg2;
typedef int Tg3;
typedef int Tg4;

extern const int Cg1;
extern const int Cg2;
extern const int Cg3;
extern int Vg1, Vg2,
Vg3, Vg4, Vg5;

extern FILE *f1;
void
common_init(void);

extern int cnt;

#endif //__Common_H__
```

### common.c:

```
#include "common.h"

const int Cg1 = 1;
const int Cg2 = 2;
const int Cg3 = 3;

int Vg1;
int Vg2;
int Vg3;
int Vg4;
int Vg5;

int cnt;

FILE *f1;

void common_init(void) {
    Vg1 = 1;
    Vg2 = 2;
    Vg3 = 3;
    Vg4 = 4;
    Vg5 = 5;

    f1 = fopen("log.txt", "w");
    fprintf(f1, "common_init started
and finished\n");
}
```

**M1.h:**

```
#ifndef __M1_H__
#define __M1_H__
void F1(void);
void F2(void);
#endif //__M1_H__
```

**M1.c:**

```
#include "M2.h"
#include "MErr.h"
#include "common.h"

void F1(void) {
    fprintf(f1, "F1
started\n");
    F4();

    Tg1 value1 = 10;

    fprintf(f1,
"Cg1=%d\n", Cg1);
    fprintf(f1,
"Tg1=%d\n", value1);

    F5();
    Err();
    fprintf(f1, "F1
finished\n");
}

void F2(void) {
    fprintf(f1, "F2
started\n");
    F5();
    F6();
    Err();
    fprintf(f1, "F2
finished\n");
}
```

**M2.h:**

```
#ifndef __M2_H__
#define __M2_H__
void F4(void);
void F5(void);
void F6(void);
#endif //__M2_H__
```

**M2.c:**

```
#include "M1.h"
#include "MErr.h"
#include "common.h"

void F4(void) {
    fprintf(f1, "F4
started\n");
    Err();
    fprintf(f1, "F4
finished\n");
}
void F5(void) {
    fprintf(f1, "F5
started\n");

    fprintf(f1,
"Cg2=%d\n", Cg2);

    Err();
    fprintf(f1, "F5
finished\n");
}
void F6(void) {
    fprintf(f1, "F6
started\n");

    fprintf(f1,
"cnt=%d\n", cnt);
    if (cnt != 0) {
        cnt--;
        F2();
    }

    Err();
    fprintf(f1, "F6
finished\n");
}
```

**M3.h:**

```

#ifndef __M3_H__
#define __M3_H__
void F3(void);
#endif // __M3_H__

```

**M3.c:**

```

#include "MErr.h"
#include "common.h"

void F7(void);
void F8(void);

void F3(void) {
    fprintf(f1, "F3
started\n");
    F7();
    F8();
    Err();
    fprintf(f1, "F3
finished\n");
}
void F7(void) {
    fprintf(f1, "F7
started\n");

    fprintf(f1,
"Cg3=%d\n", Cg3);

    Err();
    fprintf(f1, "F7
finished\n");
}
void F8(void) {
    fprintf(f1, "F8
started\n");
    fprintf(f1,
"cnt=%d\n", cnt);
    if (cnt != 0) {
        cnt--;
        F3();
    }
    Err();
    fprintf(f1, "F8
finished\n");
}

```

**MErr.h:**

```
#ifndef __MErr_H__
#define __MErr_H__
void Err(void);
#endif //__MErr_H__
```

**MErr.c:**

```
#include "common.h"

void Err(void) {
    fprintf(f1, "Err
started\n");
    fprintf(f1, "Err
finished\n");
}
```

**main.c:**

```
#include "M1.h"
#include "M3.h"
#include "common.h"

int main(void) {
    common_init();

    fprintf(f1, "main started\n");
    Vg1 = 0;

    fprintf(f1, "Vg1=%d\n", Vg1);
    fprintf(f1, "Vg2=%d\n", Vg2);
    fprintf(f1, "Vg3=%d\n", Vg3);
    fprintf(f1, "Vg4=%d\n", Vg4);
    fprintf(f1, "Vg5=%d\n", Vg5);

    F1();
    cnt = 3;
    F2();
    cnt = 3;
    F3();

    fprintf(f1, "main finished\n");
    return 0;
}
```



**Текст файлу з результуючим протоколом повідомлень роботи програми.**

common_init	F1 finished	F6 started	Err finished	Err finished	Err started
started and	F2 started	cnt=1	F6 finished	F7 finished	Err finished
finished	F5 started	F2 started	Err started	F8 started	F3 finished
main started	Cg2=2	F5 started	Err finished	cnt=2	Err started
Vg1=0	Err started	Cg2=2	F2 finished	F3 started	Err finished
Vg2=2	Err finished	Err started	Err started	F7 started	F8 finished
Vg3=3	F5 finished	Err finished	Err finished	Cg3=3	Err started
Vg4=4	F6 started	F5 finished	F6 finished	Err started	Err finished
Vg5=5	cnt=3	F6 started	Err started	Err finished	F3 finished
F1 started	F2 started	cnt=0	Err finished	F7 finished	Err started
F4 started	F5 started	Err started	F2 finished	F8 started	Err finished
Err started	Cg2=2	Err finished	F3 started	cnt=1	F8 finished
Err finished	Err started	F6 finished	F7 started	F3 started	Err started
F4 finished	Err finished	Err started	Cg3=3	F7 started	Err finished
Cg1=1	F5 finished	Err finished	Err started	Cg3=3	F3 finished
Tg1=10	F6 started	F2 finished	Err finished	Err started	Err started
F5 started	cnt=2	Err started	F7 finished	Err finished	Err finished
Cg2=2	F2 started	Err finished	F8 started	F7 finished	F8 finished
Err started	F5 started	F6 finished	cnt=3	F8 started	Err started
Err finished	Cg2=2	Err started	F3 started	cnt=0	Err finished
F5 finished	Err started	Err finished	F7 started	Err started	F3 finished
Err started	Err finished	F2 finished	Cg3=3	Err finished	main finished
Err finished	F5 finished	Err started	Err started	F8 finished	