

## Лабораторная N 1

Использование нитей в параллельном программировании.

### Задание

При помощи функций библиотеки pthread операционной системы Linux написать программу на языке "Си".

Программа должна создавать несколько нитей, каждая из которых выполняла бы свою часть задачи.

### Варианты

1. Найти сумму элементов матрицы.
2. Найти место заданного элемента в массиве.
3. Отсортировать массив.
4. Найти определитель матрицы.
5. Найти сумму элементов массива.
6. Создать массив в элементы которого являются суммой пар элементов двух других массивов.
7. Найти клетки на шахматной доске, которые не бьются ферзем.
8. Найти клетки на шахматной доске, которые бьются ферзем

### Пример выполнения

Задание: найти сумму чисел от 1 до 99.

---

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <signal.h>
#include <netdb.h>
#include <unistd.h>
#include <pthread.h>

int globs[5];

static void *handler(void *n)
{
    int nn = (int) n;
    int i, s;
    for(i = 1, s = 0; i<100; i++)
    {
        if( i % 5 == nn )
        {
            s+=i;
            printf("thread N=%d processing i=%d\n", nn, i );
        }
    }
    globs[nn] = s;
    printf("summ[%d]=%d\n", n, s );
}

main()
```

---

---

```
{
    pthread_attr_t attr;
    pthread_init();
    int i, s;
    attr = pthread_attr_new();
    pthread_attr_set(attr, PTHREAD_ATTR_NAME, "handler");
    pthread_attr_set(attr, PTHREAD_ATTR_CANCEL_STATE,
PTH_CANCEL_ASYNCHRONOUS);

    for( i = 0; i < 5; i++ )
    {
        pthread_spawn(attr, handler, (void *)i);
    }
    printf("nt=%ld\n", pthread_ctrl(PTH_CTRL_GETTHREADS));
    pthread_sleep(10);

    for( i = 0, s = 0; i < 5; i++ )
    {
        s += globs[i];
    }

    printf("summ=%d\n", s );
}
```

---

Компиляция программы

```
$ cc -o foo foo.c -lpth
```

\$ - приглашение командной строки оболочки shell

cc - компилятор

foo.c - исходный текст программы

foo - исполняемый код, генерируемый компилятором

-lpth - подключение библиотеки

Краткое описание некоторых функций библиотеки pthread

pthread\_init() инициализирует библиотеку;

pthread\_kill() убивает библиотеку;

pthread\_ctrl(запрос, аргументы) информация о нитях : число нитей, приоритет, имя.

pthread\_attr\_init(атрибуты) инициализирует атрибуты

pthread\_attr\_set(атрибуты, поле, значение)

нить = pthread\_spawn(атрибуты, функции, аргументы ) создание нити;

pthread\_suspend() зарезервировать нить

pthread\_resume() задействовать ранее зарезервированную нить

pthread\_yield() передает управление нити или планировщику

pthread\_exit(значение) завершение нити.

Лабораторная N 2

Реализовать задание первой лабораторной на ОрепМР

Лабораторная N 3

Реализовать задание первой лабораторной на MRI