

POSIX Threads

Перов Максим
кафедра РЭПИ, МФТИ(ГУ)

- Это стандарт реализации потоков выполнения
- API существует для различных ОС
- Основные языки: **C** и C++

```
#include <pthread.h> // POSIX Threads
```

```
void *func(void *arg); // прототип функции потока
```

```
int main(int argc, char **argv) {  
    pthread_t thread; // объявление описания потока  
    pthread_create(&thread, NULL, func, NULL); // инициализация потока  
    ...  
    pthread_join(thread, NULL); // ожидание окончания завершения потока  
    return 0x00;  
}
```

```
int pthread_create(pthread_t *thread, const pthread_attr_t *attr,  
    void *(*func)(void *), void *arg);
```

```
/*
```

thread - адрес переменной для описания потока;

attr - атрибуты потока;

func - указатель на функцию потока;

arg - указатель, который передаётся функции потока в качестве параметра;

```
*/
```

```
int pthread_join(pthread_t thread, const void **status_addr);
```

```
/*
```

```
    thread - описание ожидаемого потока;
```

```
    status_addr - адрес указателя данных, которые вернул ожидаемый поток;
```

```
*/
```

```
#include <pthread.h> // POSIX Threads
```

```
pthread_mutex_t mutex; // объявление мьютекса
```

```
pthread_mutex_init(&mutex, NULL); // инициализация мьютекса
```

```
...
```

```
pthread_mutex_lock(&mutex); // захват мьютекса
```

```
// работа с общей памятью
```

```
pthread_mutex_unlock(&mutex); // освобождение мьютекса
```

```
int pthread_mutex_init(pthread_mutex_t *mutex,  
    const pthread_mutexattr_t *attr);
```

```
/*  
    mutex - адрес переменной мьютекса;  
    attr - атрибуты мьютекса;  
*/
```

```
int pthread_mutex_lock(pthread_mutex_t *mutex);  
  
int pthread_mutex_unlock(pthread_mutex_t *mutex);  
  
/*  
  mutex - адрес переменной мьютекса;  
*/
```



```
gcc my_pthread_prog.c -o my_pthread_prog -lpthread
```

Разработать программу на языке Си с использованием pthread'ов, выполняющую суммирование массива из 10 млн. элементов, заполненного единицами.

Вывод результата сложения отображать на экране.