

Домашнее задание 6

Исследование ускорения программы за счет многопоточности

Написать программу, которая генерирует $N > 1000000000$ случайных чисел и подсчитывает их побитовую сумму (XOR). Программа должна аргументом командной строки принимать K – количество потоков для вычисления суммы. Работа делится равным образом между потоками, а в конце подсчитывается общая сумма и затраченное время. Исследовать получившуюся программу, запустив ее с разным количеством потоков

Deadline 2023-12-05

Как создать поток

```
int pthread_create(  
    pthread_t *thread,           // thread_id  
    const pthread_attr_t *attr,  // attributes  
    void *(*start_routine) (void *), // entrypoint  
    void *arg                    // entrypoint args  
);
```

Как дождаться завершения потока

```
int pthread_join(  
    pthread_t thread, // thread id  
    void **retval      // return status  
);
```

Значения `retval` можно вернуть из потока с помощью:

```
void pthread_exit(void *retval);
```

Как избежать состояния гонки?

Средства синхронизации будет проходить на следующем семинаре, но проблемы могут возникнуть уже сейчас. Если несколько потоков работают с одними и теми же данными без средств синхронизации, то результат работы может быть неопределен. Решение простое – не работать с одними данными.