

Ответы на вопросы задания № 5.

1. Почему при небольших значениях *sz* цикл *cilk_for* уступает циклу *for* в быстродействии?

Ответ: При выполнении цикла *cilk_for* требуется время для распределения данных по потокам. При небольших значениях *sz* такое время может превышать общее время выполнения цикла *for* в одном потоке.

2. В каких случаях целесообразно использовать цикл *cilk_for*?

Ответ: Основное требование к телу цикла *cilk_for* состоит в том, чтобы итерации цикла были независимы по данным. Такое требование накладывается, чтобы каждая итерация могла быть выполнена в параллели с любой другой итерацией. Цикл *cilk_for* целесообразно использовать в случае, когда количество итераций не менее 1000.

3. В чем принципиальное отличие параллелизации с использованием *cilk_for* от параллелизации с использованием *cilk_spawn* в паре с *cilk_sync*?

Ответ: *cilk_for* распараллеливает цикл принудительно, в то время как *cilk_spawn* обозначает точку порождения, в которой создаётся новая задача. Её выполнение может быть продолжено данным потоком или захвачено другим потоком в паре с *cilk_sync*.

cilk_sync обозначает точку синхронизации. В этой точке выполнение задач синхронизуется (барьерная синхронизация). Выполнение потоков приостанавливается до тех пор, пока не будет решена самая «медленная» задача.

Ссылка на GitHub: <https://github.com/alexleo20/lab1>