

## Задание 5

**Ответьте на вопросы: почему при небольших значениях `sz` цикл `cilk_for` уступает циклу `for` в быстродействии?**

Потому что в цикле `cilk_for` итерации распределяются между потоками . Сначала каждому потоку “раздают” итерации цикла , затем каждый поток выполняет свои итерации , а потом происходит синхронизация между потоками . Раздача итераций и синхронизация между потоками занимают некоторое небольшое время . Однако при небольшом количестве итераций из-за этого времени последовательная реализация будет быстрее (время синхронизации будет велико по сравнению с временем вычислений ) . Но при больших количествах итераций целесообразнее использовать параллельную реализацию (время синхронизации будет очень мало по сравнению с временем вычислений ) .

**В каких случаях целесообразно использовать цикл `cilk_for` ?**

При больших количествах итераций , итерации цикла не должны зависеть друг от друга и не должны содержать операторы принудительного перехода

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

`Cilk_for` распределяет итерации цикла между потоками . Конструкция `cilk_spawn` используется перед вызовом функции . Родительский поток вызывает функцию с `cilk_spawn` , которую выполняет дочерний поток параллельно родительскому