

ЛР – 6. Замки и барьеры.

Синеоков Иван

ПИН-33

Замки:

Вывод:

1 000 000:

```
Enter the size: 1000000  
Passed time: 239ms
```

10 000 000:

```
Enter the size: 10000000  
Passed time: 2310ms
```

Сравнение с другими способами:

Секциями:

1 000 000:

```
Enter the size: 1000000  
Passed time: 13ms
```

10 000 000:

```
Enter the size: 10000000  
Passed time: 117ms
```

100 000 000:

```
Enter the size: 100000000  
Passed time: 845ms
```

Последовательно:

1 000 000:

```
Консоль отладки Microsoft Visual Studio  
Enter the size: 1000000  
Passed time: 27ms
```

10 000 000:

```
Консоль отладки Microsoft Visual Studio  
Enter the size: 10000000  
Passed time: 159ms
```

100 000 000:

```
Консоль отладки Microsoft Visual Studio
Enter the size: 100000000
Passed time: 1027ms
```

Редукцией:

1 000 000:

```
Консоль отладки Microsoft Visual Studio
Enter the size: 1000000
Passed time: 11ms
```

10 000 000:

```
Консоль отладки Microsoft Visual Studio
Enter the size: 10000000
Passed time: 31ms
```

100 000 000:

```
Консоль отладки Microsoft Visual Studio
Enter the size: 100000000
Passed time: 118ms
```

Atomic:

1 000 000:

```
Enter the size: 1000000
Passed time: 32ms
```

10 000 000:

```
Enter the size: 10000000
Passed time: 222ms
```

100 000 000:

```
Enter the size: 100000000
Passed time: 2116ms
```

Примечание:

Действие замков значительно замедлило выполнение программы: очевидно их частый вызов и постоянные очереди привели к значительному снижению производительности.

Насколько мне известно, они были реализованы для синхронизации в высокой степени независимых частей кода, которым редко необходима синхронизация, или части кода из нескольких строк, логика которой не допускает распараллеливание.

Барьерную синхронизацию в своём задании мне реализовать не удалось без потери логики. Если я не правильно понял задание, или у вас есть решение этой проблемы, напишите, пожалуйста, мне, я создам и такую реализацию.