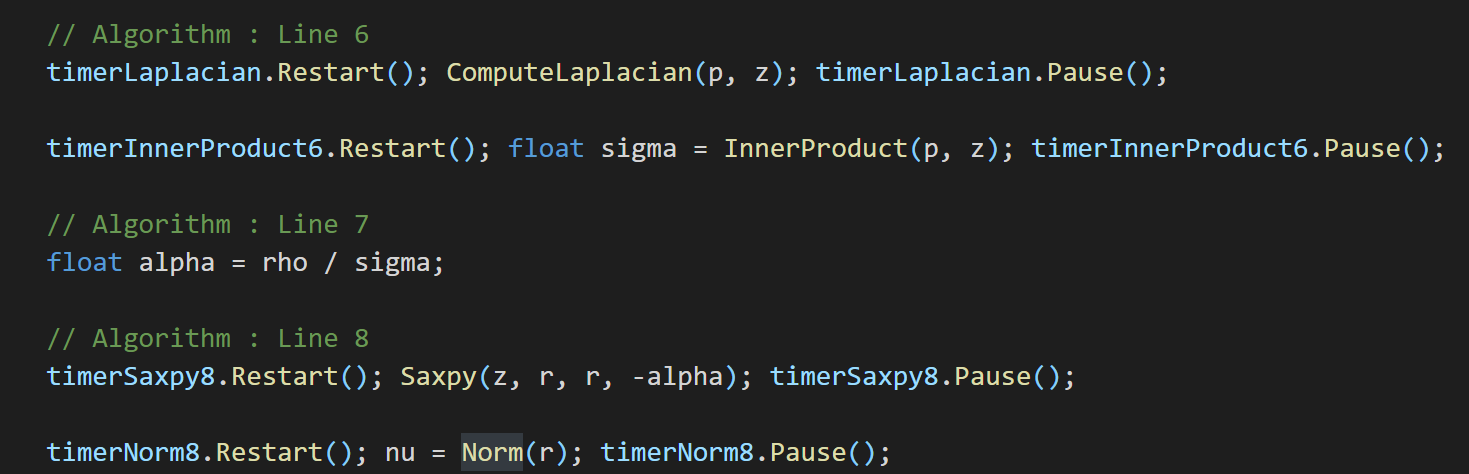
# Task 1

* Basically, I add a timer var for each kernel for each line of the algorithm, then for each time that kernel being executed, I Restart and Pause before and after that execution to record the total running time for all iterations.

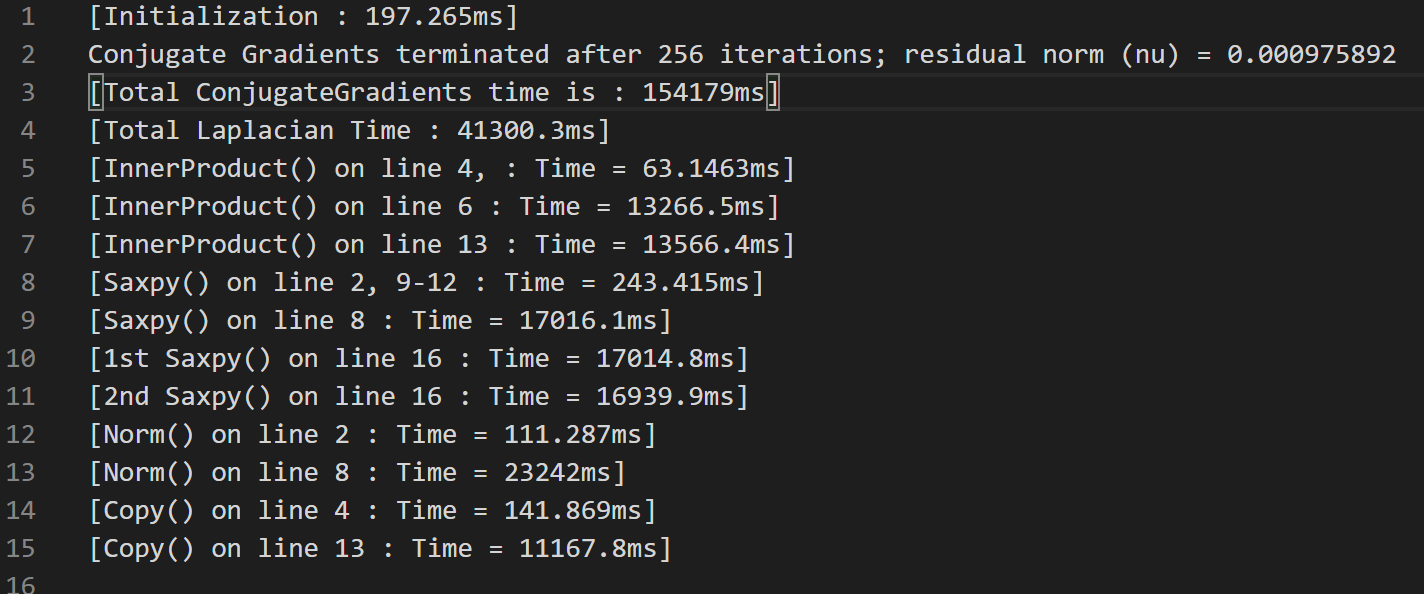
e.g.



## Parallel result:

We can calculate from the result that the sum of all individual kernel is around 10,6623ms, which is almost equal to 10,6631ms

## Single Thread Result:



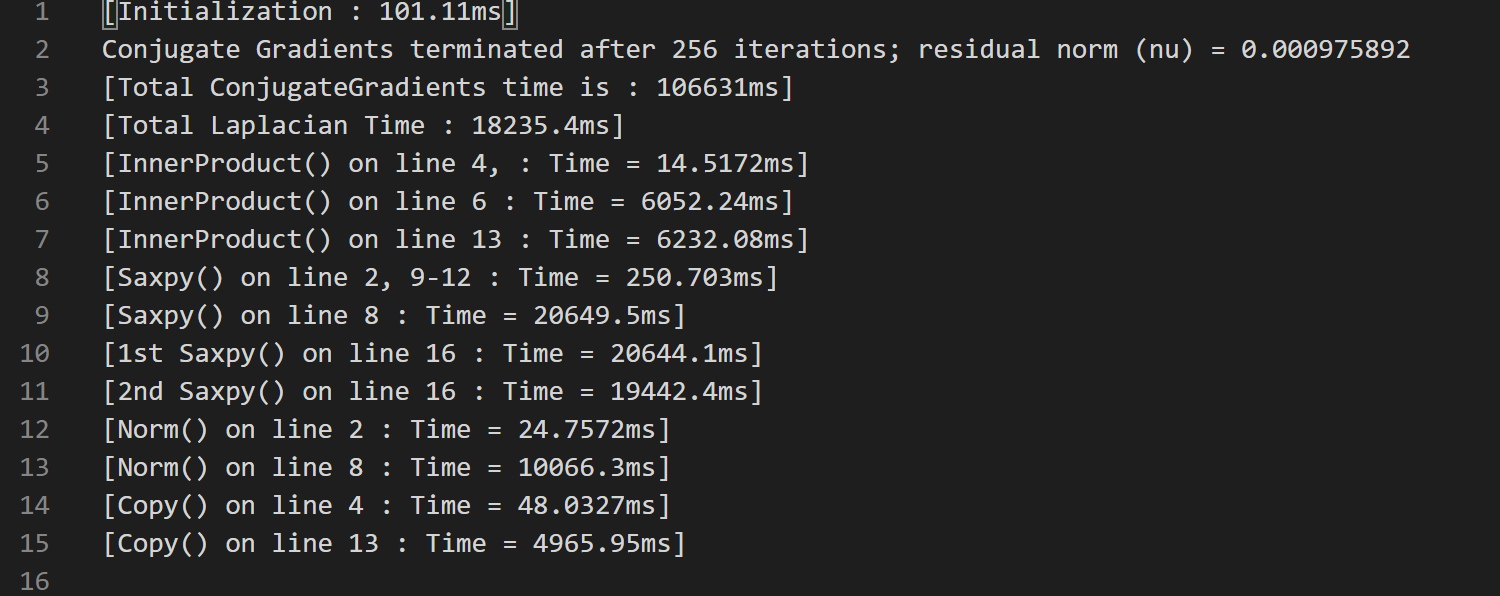
We can calculate from the result that the sum of all individual kernel is around 154,099‬ms, which is almost equal to 154,179ms

# 

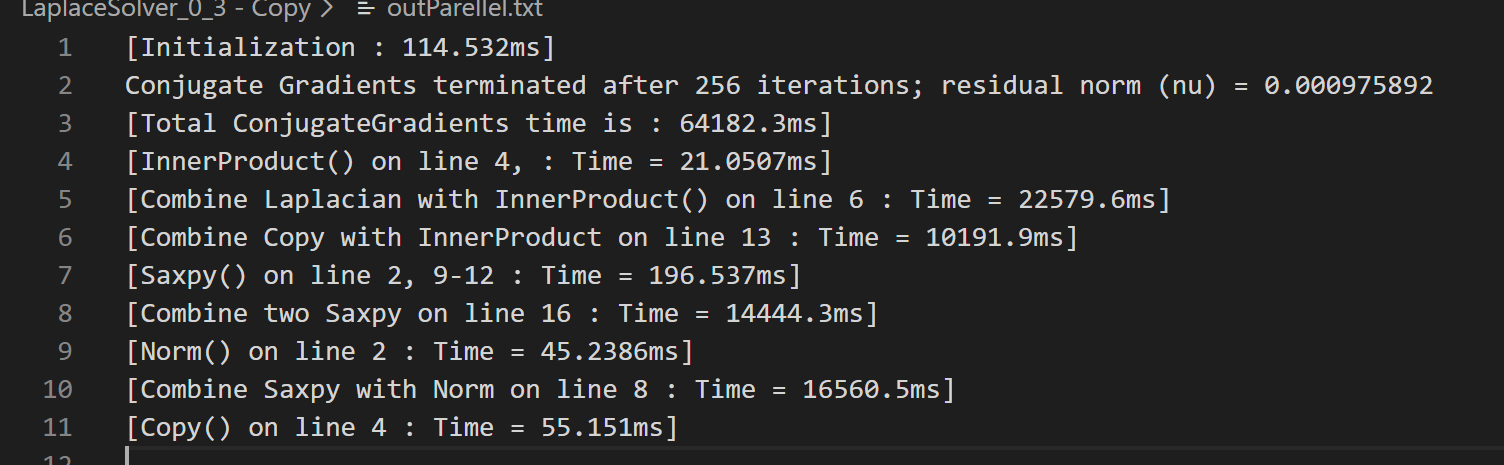
# 

# Task2

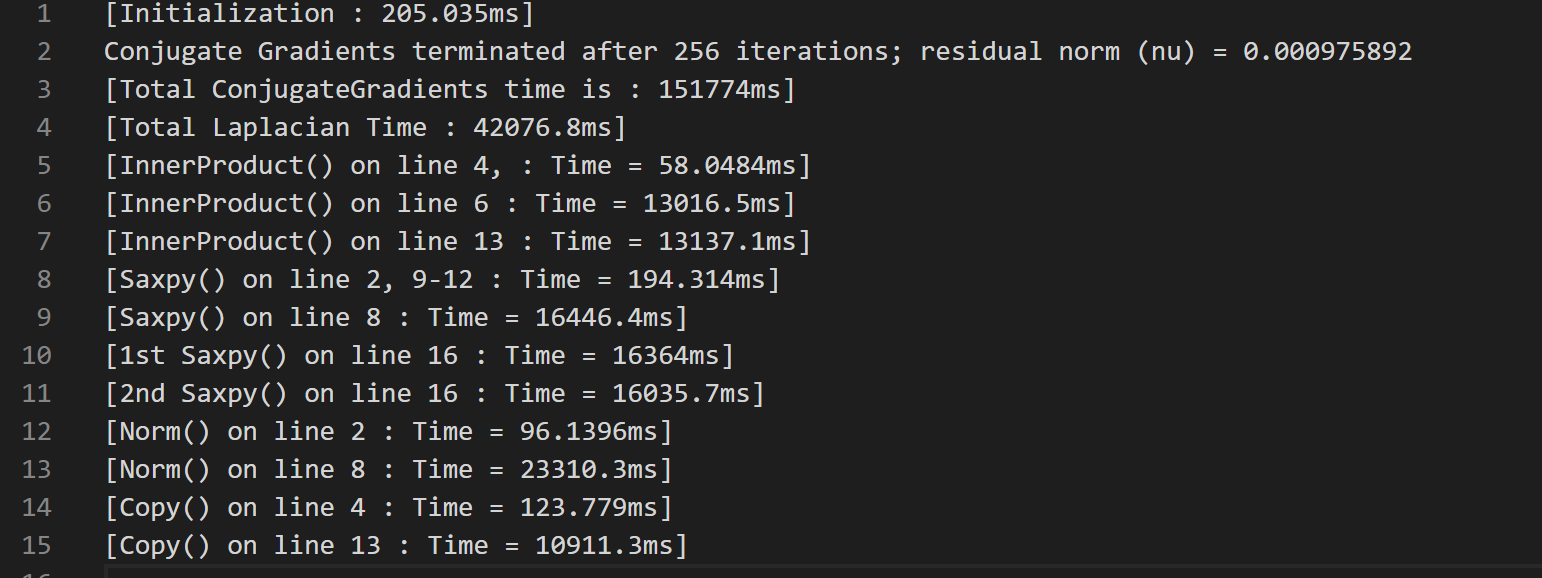
Parallel Before Combining:



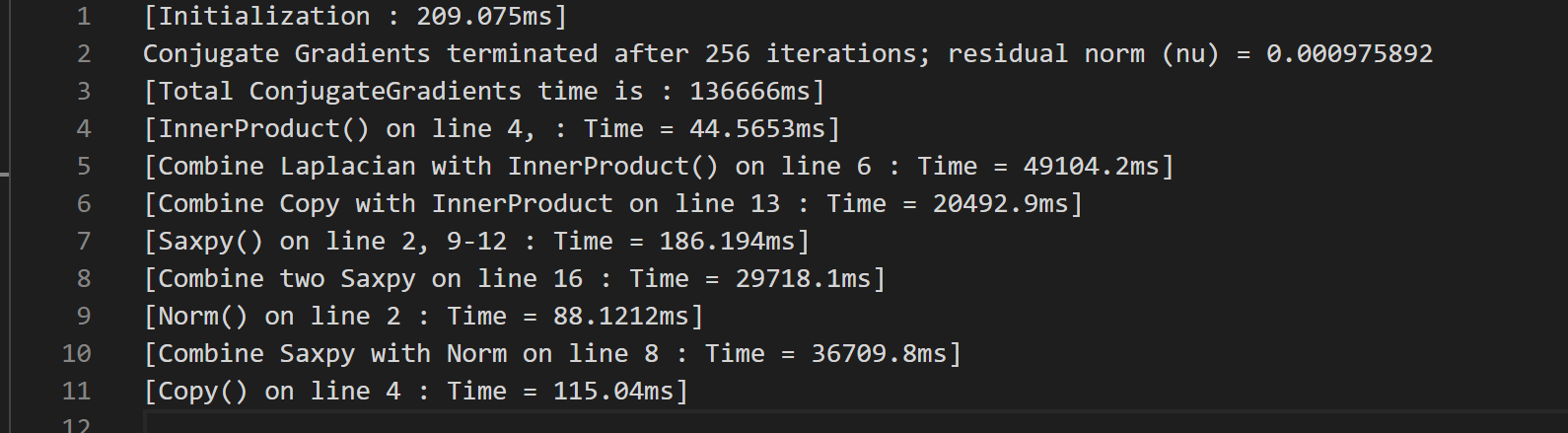
Parallel After Combining:



Single Thread Before Combing:

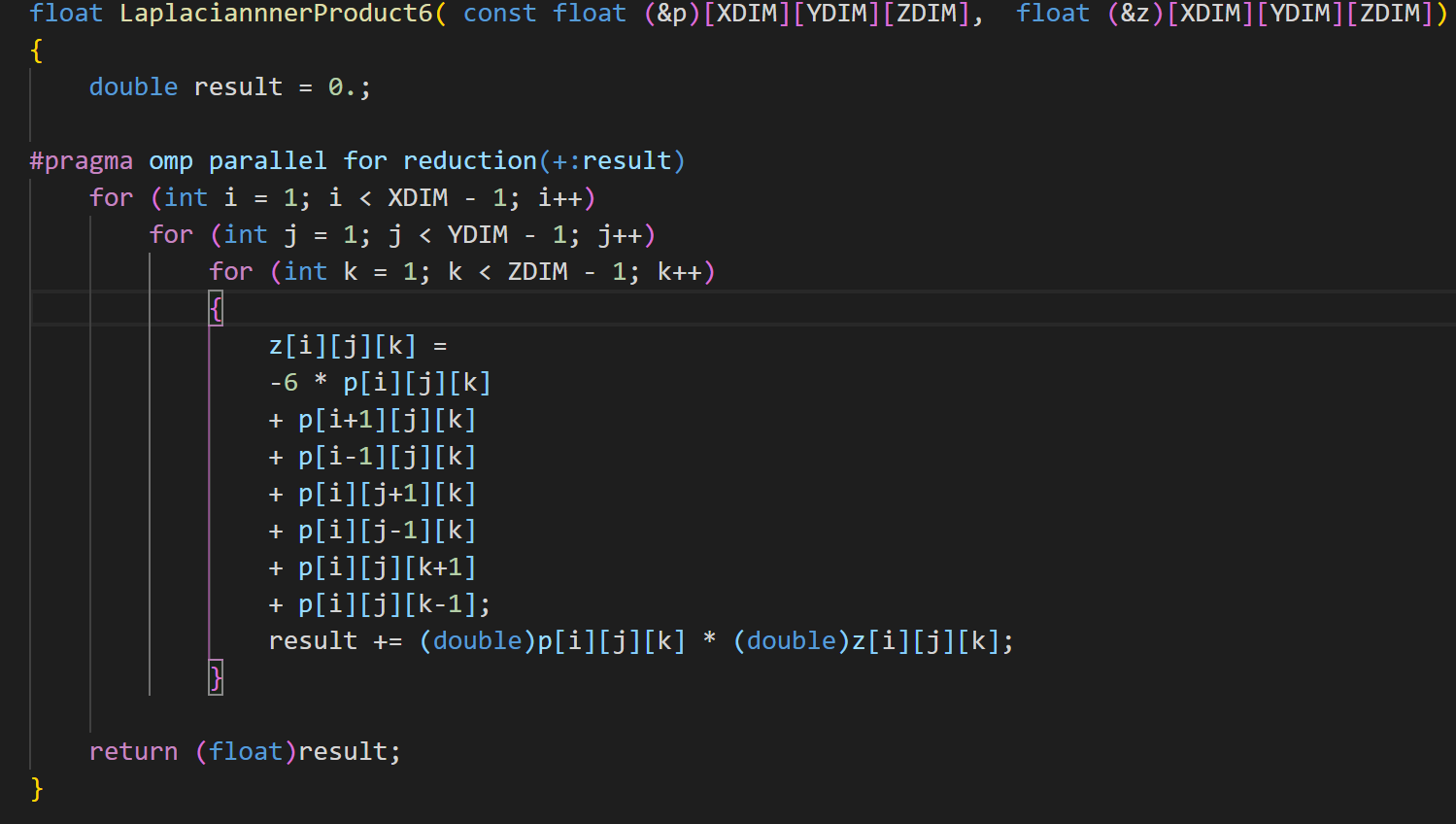


Single Thread After Combining:



**In general, my implementation of merging the kernel reduces total parallel running time from 106631ms to 64182ms. However, it does not give noticeable improvement for any single thread test.**

## Combine Laplacian with InnerProduct() on line 6:



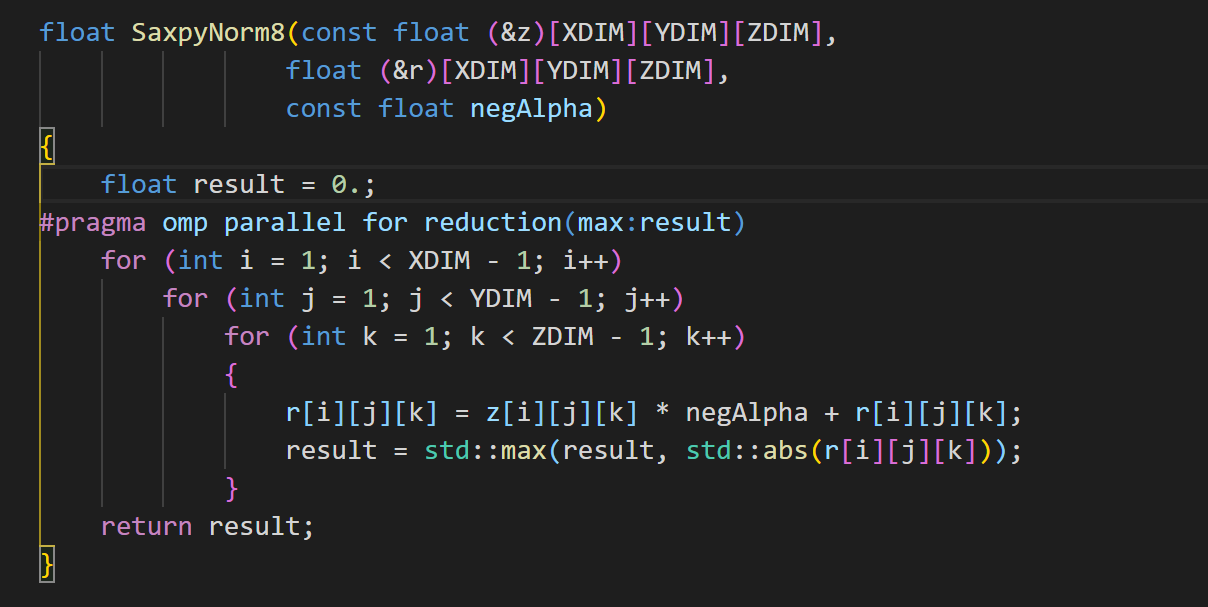
### Parallel:

We can see from the result that the running time for line 6 reduce from **18235+6052=24287ms** to **22579.3ms.** It is not noticeable, probably because the memory access of Laplacian make InnerProduct needed to be wait until the z[i][j][k] is calculated, during which the CPU does not make full use.

### Single Thread:

We can see from the result that the running time for line 6 reduce from **13016+42076=55092ms** to **49104.2ms,**

## Combine Saxpy and Norm on line8:



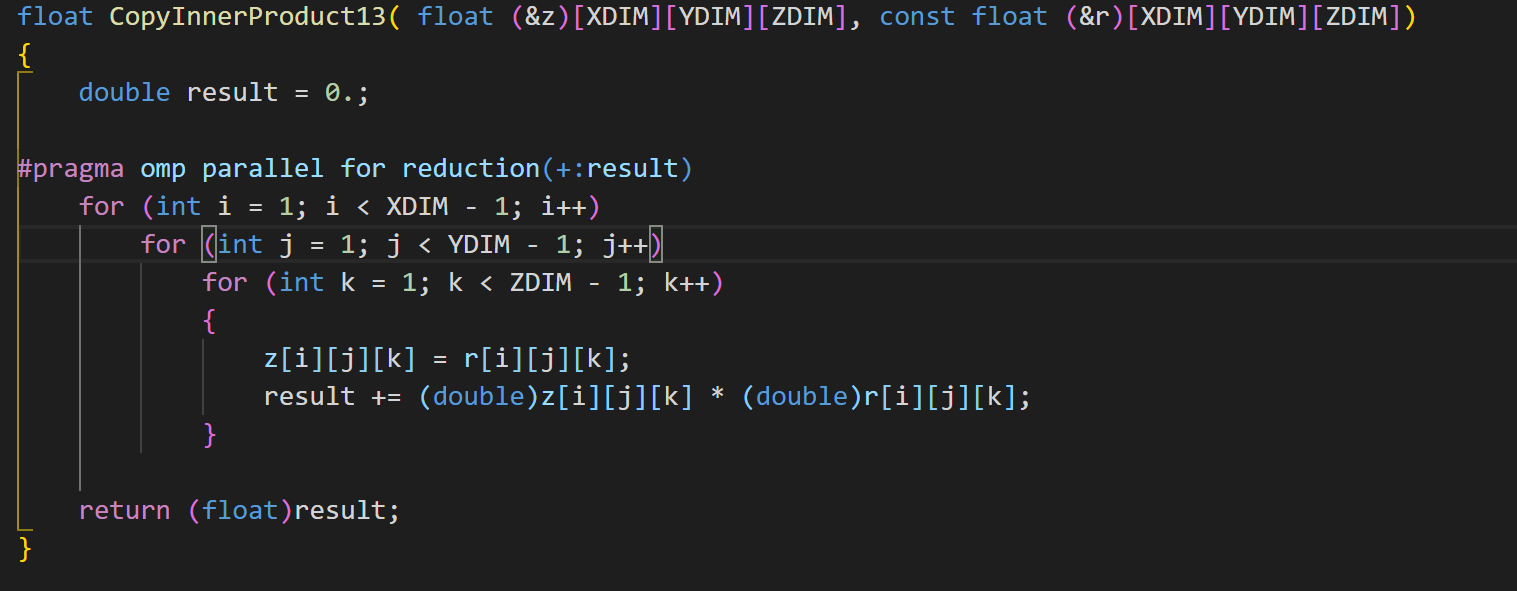
### Parallel:

We can see the running time for line8 reduce greatly from **20649+10066=30715ms** to **16560ms**

### Single Thread:

We can see the running time for line8 change from **16446+23310=39756ms** to **36709ms**

## Combine Copy and InnerProduct for line 13:



### Parallel:

We can see the running time for line13 reduce from **6232+4965=11197** to **10191ms.**

It is not noticeable, probably because the InnerProduct needed to be wait until the z[i][j][k] is copyed, during which the CPU does not make full use.

### Single:

We can see the running time for line13 reduce from **13137+10911=24048ms** to **20492ms**

## Combine Saxpy on line 16

### Parallel:

We can see that the Saxpy on line 16 runtime is greatly reduced from

**(20644+19442) = 40106ms** to **14444ms**

### Single Thread:

We can see the running time for line16 reduce from **16364+16035=32399ms to 29718ms**