HW 11

Find the isoefficiency of the following program under the assumptions given below:

```
r = 0;
for (i=0; i < n; i++) r = r + a[i];
```

When the recurrence is written as:

program 1:

```
double r[threadCount];
s = 0;
#pragma omp parallel for
for (i=0; i < n; i++)
  r[omp_get_thread_num( )] = r[omp_get_thread_num( )] + a[i];

for (i=0; i < threadCount; i++)
  s += [i];
```

When the recurrence is written as:

program 2:

```
#pragma omp parallel for reduction(+:r)
for (i=0; i < n; i++)
  r = r + a[i];
```

When the recurrence is written as:

program 3:

```
#pragma omp parallel for simd reduction(+:r)
for (i=0; i < n; i++)
  r = r + a[i];
```

In program 3, assume that it is like the program in 2 except that four adds are done for the cost of one add when finding the partial sums on each thread, and when combining the partial sums across threads four partial sums can be added each time.

Compare the work involved in the recurrence of 1, 2 and 3 at 4 threads and at 256 threads using the iso-efficiency relationships you came up with as answers.