

Array sum

1 Description of the problem

Let's assume that we have an array `x` of size `n` that contains double-precision (double) real values. We want to compute the sum of all its coefficients. For that, we have a `superfast_sum` routine available which is super-optimized but sequential. We assume that the `superfast_sum` routine can be used, but its code is not available.

This exercise is about parallelizing the array sum operation while using as much as possible the `superfast_sum` sequential routine.

2 Package content

In the `array_sum` directory you will find the following files:


- `main.c`: this file contains the main program which first initializes the `x` array and, then, calls the `superfast_sum` and `superfast_sum_par` routines. Then, it checks that the two return the same result. **Only this file has to be modified for this exercise.**
- `aux.c`, `aux.h`: these two files contain auxiliary routines and **must not be modified.**

The code can be compiled with the `make` command: just type `make` inside the `array_sum` directory; this will generate a `main` program that can be run like this:

```
$ ./main n
```

where `n` is the size of the `x` array.

3 Assignment

-  At the beginning, the `superfast_sum_par` routine simply make a call to `superfast_sum` on the whole `x` array. Modify this routine in order to parallelize the array sum operation while using, as much as possible, the `superfast_sum` routine.