Tips & tricks - Week 7

Task 1

- Remember the clauses required for compiling for loops with OpenMP provided in the lecture notes:
  - The loop can not contain break, return, exit statements.
  - The continue statement is allowed.
  - The index update has to be an increment (or decrement) by a fixed amount.
  - The loop index variable is automatically private, and changes to it inside the loop are not allowed.
- OpenMP's Data Environment Documentation might be a good read if you are wondering about what directives you can use when parallelize the funcitons

Task 2

- The different schedulers you can use are `static`, `dynamic`, `guided`, `auto`, `runtime`
- Scheduling with chunksize provided is done by `schedule(type, chk_size)` in the `omp` directive
- OpenMP comes with tools for timing parallelized code. Use `omp_get_wtime()` to get time in seconds.

```
double start = omp_get_wtime();
// Do some parallel work here
double end = omp_get_wtime();
double total = end - start;
```

Task 3

- Note: The matrix is not a pointer-to-pointer `double **int`, but just `double *int`. We index in 2d by `i*n + j` instead.
  - You could use a simple defined macro to help with the linear indexing `#define idx(i,j) (i*n + j)`