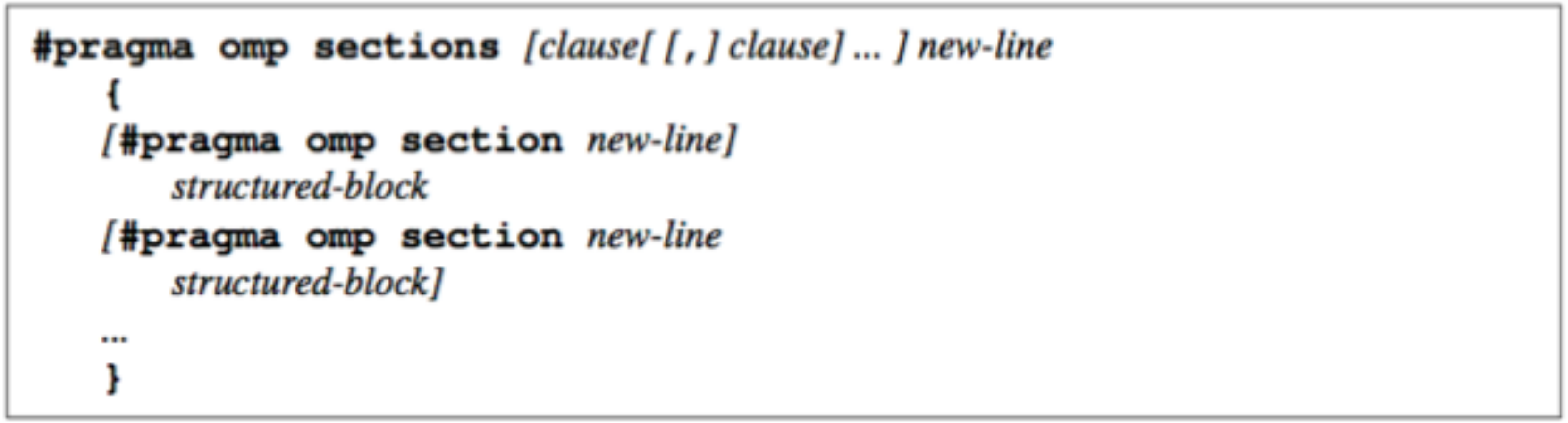
Additional Coordination Constructs

Barrier, Single, and Master Directives

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Parallel Sections

- Provide another way of creating a team of threads
  - In addition to construct parallel or parallel for
- From the Openmp 4.5 standard:



```
#pragma omp sections [clause[ [,] clause] ... ] new-line
    {
    [#pragma omp section new-line]
        structured-block
    [#pragma omp section new-line
        structured-block]
    ...
    }
```

- Independent different pieces of code assigned to different threads
- (BTW: openmp standard is the (semi-)final arbiter
  - http://www.openmp.org/specifications/
  - Final arbiter is of course your compiler … hopefully it implements the latest standard … check always

`barrier` Construct: Making Everyone Wait

- This can be thought of as an event synchronization construct

```
#pragma omp barrier
```

- No thread can pass the barrier directive unless all threads (in the current team) have arrived at it
- The programmer must take care to ensure all threads (in the team) encounter this statement or none of them do, for every execution of the program

The `master` Construct

- In a parallel region, sometimes you want some action to be done only by the master thread
  - The parallel region may be a “parallel for” or a “parallel” construct, for example
- Syntax:

```
#pragma omp master
    structured_block
```

- The master thread executes the structured_block, while
- All the other threads pass past it
  - I.e., they do not execute the structured_block nor do they wait for the master thread to execute it

The `single` Construct

- Similar in spirit to the master construct
- In a parallel region, sometimes you want some action to be done only by a single thread
  - It doesn't matter which thread executes it
- Syntax:

```
#pragma omp single
    structured_block
```

- The first thread to arrive at this directive executes the structured_block, while
- All the other threads pass past it
  - I.e., they do not execute the structured_block nor do they wait for execution of this structured block by the first thread