OpenMP Clauses (5A)

Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using LibreOffice.

Based on

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

Data Sharing Attribute Clauses

Data-sharing attribute clauses apply only to variables whose names are visible in the construct on which the clause appears.

default(shared | none)
shared(list)
private(list)
firstprivate(list)
lastprivate(list)
linear(list[:linear-step])

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

default

default(shared | none)

Explicitly determines the default data-sharing attributes of variables that are referenced in a parallel, task, or teams construct, causing all variables referenced in the construct that have implicitly determined data-sharing attributes to be shared.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

shared

shared(list)

Declares one or more list items to be shared by tasks generated by a parallel, task, or teams construct. The programmer must ensure that storage shared by an explicit task region does not reach the end of its lifetime before the explicit task region completes its execution.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

private

private(list)

Declares one or more list items to be private to a task or a SIMD lane.
Each task that references a list item that appears in a private clause
in any statement in the construct receives a new list item.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

firstprivate

firstprivate(list)

Declares list items to be private to a task, and initializes each of them with the value that the corresponding original item has when the construct is encountered.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

lastprivate

lastprivate(list)

Declares one or more list items to be private to an implicit task or to a SIMD lane, and causes the corresponding original list item to be updated after the end of the region.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

linear

linear(list[:linear-step])

Declares one or more list items to be private to a SIMD lane and to have a linear relationship with respect to the iteration space of a loop.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

reduction

reduction(reduction-identifier:list)Specifies a reduction-identifier and one or more list items. The reduction-identifier must match a previously declared reduction-identifier of the same name and type for each of the list items.

Operators for reduction (initialization values)

+ (0)	**|** (0)
***** (1)	**^** (0)
- (0)	**&&** (1)
& (~0)	**||** (0)

max(Least representable number in reduction list item type)

min(Largest representable number in reduction list item type)

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

Data Copying Clauses

copyin

copyprivate

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

Copyin

copyin(list)

Copies the value of the master thread's threadprivate variable to the threadprivate variable of each other member of the team executing the parallel region.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

copyprivate

copyprivate(list)

Broadcasts a value from the data environment of one implicit task to the data environments of the other implicit tasks belonging to the parallel region.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

Map Clauses

map

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

Map Clauses

map([map-type:]list)

Map a variable from the task's data environment to the device data environment associated with the construct.

map-type:

alloc: On entry to the region each new corresponding list item has an undefined initial value.

to: On entry to the region each new corresponding list item is initialized with the original list item's value.

from: On exit from the region the corresponding list item's value is assigned to each original list item.(Continued >)

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

SIMD

safelen

collapse

simdlen

aligned

uniform

inbranch

notinbranch

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

safelen

safelen(length)

If used then no two iterations executed concurrently with SIMD instructions can have a greater distance in the logical iteration space than its value.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

collapse

collapse(n)

A constant positive integer expression that specifies how many loops are associated with the loop construct.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

simdlen

simdlen(length)

A constant positive integer expression that specifies the number of concurrent arguments of the function.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

aligned

aligned(argument-list[:alignment])

Declares one or more list items to be aligned to the specified number of bytes. alignment, if present, must be a constant positive integer expression. If no optional parameter is specified, implementation-defined default alignments for SIMD instructions on the target platforms are assumed.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

aligned

aligned(argument-list[:alignment])

Declares one or more list items to be aligned to the specified number of bytes. alignment, if present, must be a constant positive integer expression. If no optional parameter is specified, implementation-defined default alignments for SIMD instructions on the target platforms are assumed.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

uniform

uniform(argument-list)

Declares one or more arguments to have an invariant value for all concurrent invocations of the function in the execution of a single SIMD loop.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

inbranch

inbranch

Specifies that the function will always be called from inside a conditional statement of a SIMD loop.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

notinbranch

notinbranch

Specifies that the function will never be called from inside a conditional statement of a SIMD loop.

https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf

References

[1] ftp://ftp.geoinfo.tuwien.ac.at/navratil/HaskellTutorial.pdf

[2] https://www.umiacs.umd.edu/~hal/docs/daume02yaht.pdf