

Padreando-el-BP2.pdf



BlackTyson



Arquitectura de Computadores



2º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada

Las cláusulas acompañan a las directivas para darles un motivo.

Las directivas que admiten cláusulas son `parallel` y las directivas de trabajo compartido, todas las demás no aceptan cláusulas.

Cláusulas de las directivas:

- `Parallel`: `if`, `num_threads(num)`, `shared`, `private`, `firstprivate`, `default(num)`, `reduction`, `copyin`
- `Do/For`: `private`, `lastprivate`, `firstprivate`, `reduction`, `schedule`, `ordered`, `nowait`.
- `sections`: `private`, `lastprivate`, `firstprivate`, `reduction`
- `single`: `private`, `firstprivate`.
- `parallel do/for`: `if`, `num_threads(num)`, `shared`, `private`, `lastprivate`, `firstprivate`, `default(num)`, `reduction`, `copyin`, `schedule(num)`, `ordered(num)`
- `parallel sections`: `if (num)`, `num_threads(num)`, `shared`, `private`, `lastprivate`, `firstprivate`, `default(num)`, `reduction`, `copyin`.

Las variables fuera de una región y las dinámicas son compartidas por los threads de la región. Las declaradas dentro son privadas. Esto ocurre siempre excepto con los índices de los bucles `for` (privados por determinado) y las declaradas `statics`.

Cláusulas:

- `shared(list)`:
 - Las variables de `list` se comparten por todos los threads.
 - Hay que tener cuidado con la dependencia de datos RAW.
- `private(list)`:
 - cada subproceso tiene su propia instancia de la variable
 - El valor de entrada y salida está indefinido aunque se declare fuera.
- `lastprivate(list)`
 - Combina la cláusula `private` con el valor de la última variable de la lista
 - Crea instancias privadas de una variable
- `firstprivate(list)`
 - Combina `private` con la inicialización de las variables de la lista al entrar en la región.
 - Asigna el valor de la variable a todas las hebras.
- `default()`
 - Pueden ser `none` o `shared`. `None`: el programador debe especificar el alcance las variables usadas
 - Se pueden excluir variables de ámbito usando `shared`, `private`, `firstprivate`, `lastprivate`, `reduction`.
 - Solo puede haber 1 `default`
- `reduction(operator: list)`
 - Los operadores disponibles son: `+`, `-`, `*`, `&`, `|`, `^`, `&&`, `||`
 - Reduce a un único valor usando un operador
- `copyprivate(list)`
 - Solo puede usarse con `single`. Permite que una variable privada de un thread haga difusión con variables del mismo nombre de otros threads.
 - Útil para la lectura de entrada de variables.

Ejercicios que pueden caer:

1. **Qué directivas no admiten cláusulas:** atomic, critical, master.
2. **Cuál es la función de las cláusulas:** ajustar el comportamiento de las directivas.
3. **Si queremos que el valor de una hebra single se comparta con el resto de hebras del mismo nombre usaremos:** copyprivate.
4. **Qué cláusulas crean instancias de una variable con un valor indefinido:** private y lastprivate.
5. **Si tenemos un bucle con la directiva parallel for y la cláusula reduction, y un valor n inicializado fuera, el cual va aumentando en función al operador de la cláusula(reduction(+:n), cuanto valdría n:** depende del número de iteraciones
6. **Si queremos actualizar en un bucle una variable n declarada fuera del bucle, que incrementa en función de i, que es iteradora del bucle, y el bucle es un bloque parallel for default(none), que ocurre con la variable n:** none nos obliga a especificar el alcance de n ya que no es variable iteradora ni threadprivate.
7. **Si en el anterior caso, estuviéramos ante un default(shared) y tuviéramos la variable inicializada, se obtendría un resultado correcto?:** si