Padreando-el-BP1.pdf

BlackTyson

Arquitectura de Computadores

2º Grado en Ingeniería Informática

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

BP1:
Sintaxis: #pragma omp(necesario) + nombre de la directiva (necesario) + cláusula(opcional) + newline(necesario)
La API de programación de openOMP está formado por las directivas, las funciones y las variables de entorno.
Las directivas solo se tienen en cuenta cuando se compilan con -fopenmp
Region: código encontrado en una instancia de la construcción de la subrutina de openmp
OMP_NUM_THREADS: fija el número de threads que se usan en una ejecución.
Directivas:

- Parallel(Bloque estructurado): es creada por una thread master y tiene una barrera implícita al final. Cada thread ejecuta todo el código de la región, por lo tanto no reparte tareas.
- Trabajo compartido:
    - Sections(Bloque estructurado):
        - Distribuye trozos de código independientes entre las threads.
        - Tiene una barrera implícita al final.
        - El mapeo lo hace la herramienta.
        - Presenta paralelismo a nivel de función y una estructura explícita.
    - Worksharing(bloque estructurado): fortran.
    - Single (bloque estructurado):
        - Uno de los threads ejecuta parte del código secuencial.
        - Tiene barrier implícita al final.
    - For(bucles):
        - Distribuye iteraciones de los bucles entre los threads.
        - Tiene una barrera implícita al final.
        - El mapeo lo hace la herramienta a no ser que se utilice “schedule”.
        - Para usarse se tiene que saber el número de iteraciones las cuales se deben de poder paralelizar.
        - Presenta un paralelismo a nivel de bucle y una estructura implícita.
- Combinación parallel con trabajo compartido:
    - No admite nowait.
    - Menos legible y peores prestaciones.
- Directivas comunicación y sincronización:
    - Barrier(autónoma): Punto de código donde los threads esperan a que terminen todos sus hermanos.
    - Critical(bloque estructurado):
        - Evita que varios threads entren a variables compartidas a la vez.
        - Se puede usar name para evitar deadlock
        - No tiene barrera implícita
    - Atomic(simple):
        - Alternativa a critical más eficiente.
        - No puede usarse en más de una instrucción.
        - Carece de barrera implícita final
    - Master(bloque estructurado):

- No tiene ninguna barrera implícita.
- Hace que solo la hebra máster ejecute, haciendo que varíen resultados esperados.
- Si se usa en un programa paralelo, el resultado será 0 ya que no es una herramienta de trabajo compartido.
- No ocurre sincronización de hebras.

Algunas preguntas de interés:

1. **Qué resultado arroja master cuando se hace una ejecución en paralelo:** 0.
2. **Qué directivas no tienen una barrera implícita:** master, critical, atomic.
3. **Qué componentes definen la API de programación:** las directivas, las funciones y las variables de entorno
4. **Cuantas veces se ejecuta un cout dentro de un parallel y con el límite de hebras limitado a 3 (OMP_NUM_THREADS = 3):** 3
5. **Y el mismo caso usando una directiva single:** 1.
6. **Si queremos que las hebras ejecuten 1 instrucción en exclusión mutua, que directiva usaremos:** atomic.
7. **Y si hubiera más instrucciones:** critical.
8. **Cual de las dos directivas dichas anteriormente es más eficiente:** atomic
9. **Cuantas veces se ejecuta un cout dentro de un critical y con el límite de hebras limitado a 3 (OMP_NUM_THREADS = 3):** 3
10. **Y si fuera single:** 1
11. **Que directiva usaremos si queremos crear un bloque de código que pueda ser ejecutado por hebras en paralelo:** master
    ¿Te imaginas master? es literalmente lo que **no** puede hacer -> parallel.
12. **Es más legible una instrucción for y una instrucción parallel si las combinamos:** no.
13. **Es do-while una directiva bucle de pragma OMP:** no.
14. **Qué directivas podemos encontrar en fortran:** do y workshare
15. **Qué tipo de paralelismo encontramos en una directiva sections:** nivel de funciones.
16. **Y en una directiva for:** nivel de bucle