

SWAD-plataforma-de-apoyo-a-la-do...



BrokenQuagga



Arquitectura de Computadores



2º Grado en Ingeniería Informática



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

Aprende Inglés Con nuestros cursos **GRATUITOS** para desempleados

 Junta de Andalucía
Consejería de Empleo, Formación
y Trabajo Autónomo

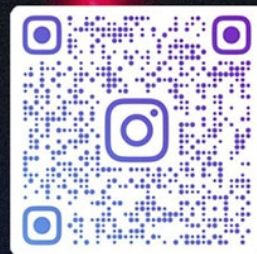
**Fórmate con nuestros cursos de Inglés
para las titulaciones A1 B1 A2 y B2
100% subvencionados para desempleados**



**Comienza
Ahora!**



**ESCANEA Y PARTICIPA EN EL
SORTEO DE UN ALIENWARE**



**LA PRIMERA RESIDENCIA GAMING
EN EL MUNDO ABRE EN MADRID**



GAMING RESIDENCES

gamingresidences.com

info@gamingresidences.com



LA PRIMERA RESIDENCIA GAMING EN
EL MUNDO ABRE EN MADRID
ESCANEA EL CÓDIGO QR Y PARTICIPA
EN EL SORTEO DE UN ALIENWARE



14/5/2021

SWAD: plataforma de apoyo a la docencia / UGR /

Examen BP3 - Grupo A2



Universidad de Granada - Grado en Ingeniería Informática
Arquitectura de Computadores

Desconocido:



Inicio:

Final: Hoy, viernes, 09:54:28

Preguntas: 10

Respuestas
válidas:

Puntuación:

Nota:

1
Elección única

Analiza el código mostrado a continuación e indica qué habría que cambiar para que se imprima la siguiente salida. Cuando `OMP_NUM_THREADS = 4`

```
int i, n = 3;  
#pragma omp parallel for private(n)  
for (i = 0; i < omp_get_max_threads(); ++i)  
    printf("Thread %d imprime: %d", omp_get_thread_num(), i+n);
```

Salida por pantalla:

```
Thread 0 imprime: 3  
Thread 1 imprime: 4  
Thread 2 imprime: 5  
Thread 3 imprime: 6
```

Usuario Profesores

- a) Cambiar private por firstprivate
- b) Cambiar private por copyprivate
- c) No hay que cambiar nada
- d) Cambiar private por lastprivate

2
Elección única

¿Cuál es la función de la cláusula `if` en el siguiente código?

```
#pragma omp parallel if(n>20)
```

Usuario Profesores

- a) No ejecutar el código del bloque estructurado si `n <= 20`
- b) Ejecutar las ramas `if` y `else` del bloque estructurado en paralelo
- c) Evitar la sobrecarga introducida al paralelizar el código para tamaños del problema pequeños
- d) Las otras respuestas no son correctas

3
Elección única

¿Con cuántas hebras se ejecuta este código si previamente se ha fijado la variable de entorno `OMP_NUM_THREADS=8`

```
omp_set_num_threads(4);  
#pragma omp parallel num_threads(2)  
{  
    f("h ll \")  
}
```

printf("hello\n");
Usuario Profesores

- ☐ a) 2
- ☐ b) 8
- ☐ c) 4
- ☐ d) 1

4
Elección única

En el siguiente fragmento de código, ¿cuántas hebras están ejecutando la región paralela?

```
long sum = 0, N=10, a[10], b[10], c[10];  
#pragma omp parallel  
{  
    int ithread = omp_get_thread_num();  
    int nthread = omp_get_num_threads();  
    #pragma omp sections  
    {  
        #pragma omp section  
        for (long i = 0; i < N; i += nthread)  
            c[i] = a[i] + b[i];  
  
        #pragma omp section  
        for (long i = ithread; i < N; i += nthread)  
            c[i] = a[i] + b[i];  
    }  
}
```

Usuario Profesores

- ☐ a) Las que indique la función omp_get_thread_num()
- ☐ b) El número de hebras posible será siempre igual al número de procesadores lógicos que tenga la máquina donde se ejecuta el código.
- ☐ c) 2
- ☐ d) N

5
Elección única

¿Cuál de los siguientes métodos para determinar el número de hebras que ejecutarán la siguiente región paralela es el más prioritario?

Usuario Profesores

- ☐ a) La función omp_set_num_threads
- ☐ b) La cláusula if
- ☐ c) La cláusula num_threads
- ☐ d) La variable de entorno OMP_NUM_THREADS

6
Elección única

Indica qué reparto de iteraciones a hebras es correcto suponiendo 2 hebras y la cláusula schedule(guided,3)

Usuario Profesores

- ☐ a)

iteración	0	1	2	3	4	5	6	7	8	9	10	11
hebra	0	0	0	1	1	1	0	0	0	1	1	1
- ☐ b)

iteración	0	1	2	3	4	5	6	7	8	9	10	11
hebra	0	0	0	0	0	0	1	1	1	0	0	0
- ☐ c)

iteración	0	1	2	3	4	5	6	7	8	9	10	11
hebra	0	0	0	0	0	0	0	1	1	0	0	0
- ☐ d)

i	i	0	1	2	3	4	5	6	7	8	9	10	11
hebra	0	0	0	0	0	0	0	0	0	0	0	0	0

iteración	0	1	2	3	4	5	6	7	8	9	10	11
hebra	0	0	0	0	0	0	1	1	1	0	0	1

7 Indica cuál de las siguiente opciones obtendrá mejores prestaciones para multiplicar una matriz triangular por un vector
Elección única Usuario Profesores

- ☐ a) `#pragma omp for private(j) schedule(static)`
`for (i=0 ; i<N ; i++) {`
`v2[i] = 0;`
`for (j=0 ; j<N ; j++)`
`v2[i] += M[i][j] * v1[j];`
- ☐ b) `#pragma omp for schedule(guided)`
`for (i=0 ; i<N ; i++) {`
`v2[i] = 0;`
`for (j=0 ; j<=i ; j++)`
`v2[i] += M[i][j] * v1[j];`
`}`
- ☐ c) `#pragma omp for private(j) schedule(guided)`
`for (i=0 ; i<N ; i++) {`
`v2[i] = 0;`
`for (j=0 ; j<=i ; j++) {`
`#pragma omp critical`
`v2[i] += M[i][j] * v1[j];`
`}`
`}`
- ☐ d) `#pragma omp for private(j) schedule(guided)`
`for (i=0 ; i<N ; i++) {`
`v2[i] = 0;`
`for (j=0 ; j<=i ; j++)`
`v2[i] += M[i][j] * v1[j];`
`}`

8 Las variables de control internas de OpenMP
Elección única Usuario Profesores

- ☐ a) Sólo se pueden modificar mediante el uso de las funciones que proporciona el API de OpenMP
- ☐ b) Pueden ser accedidas directamente por el programador
- ☐ c) Las otras respuestas son todas incorrectas
- ☐ d) Sólo se pueden modificar mediante el uso de variables de entorno en la consola del sistema

9 ¿Cómo se puede modificar el reparto de iteraciones del bucle de una directiva `#pragma omp for` entre las hebras si usamos la cláusula `schedule(runtime)` ?
Elección única Usuario Profesores

- ☐ a) Usando la variable de entorno `OMP_SCHEDULE` y la función `omp_set_schedule()`
- ☐ b) Usando sólo la función `omp_set_schedule()`
- ☐ c) Usando la variable de entorno `OMP_SCHEDULE` y la función `omp_set_schedule()`
- ☐ d) Usando sólo la variable de entorno `OMP_SCHEDULE`

10 Indica qué reparto de iteraciones a hebras es correcto suponiendo 3 hebras y la cláusula `schedule(dynamic,2)` .
Elección única Usuario Profesores

- ☐ a)

iteración	0	1	2	3	4	5	6	7	8	9
hebra	0	0	1	1	2	2	2	2	0	1



LA PRIMERA RESIDENCIA GAMING EN
EL MUNDO ABRE EN MADRID

ESCAÑEA EL CÓDIGO QR Y PARTICIPA
EN EL SORTEO DE UN ALIENWARE



b)

iteración	0	1	2	3	4	5	6	7	8	9
hebra	0	0	0	1	1	1	2	2	2	0



c)

iteración	0	1	2	3	4	5	6	7	8	9
hebra	0	0	1	1	2	2	0	0	0	2

•



d)

iteración	0	1	2	3	4	5	6	7	8	9
hebra	0	0	1	1	2	2	1	1	0	0

gamingresidences.com
info@gamingresidences.com