



# Examen BP3



Universidad de Granada - Doble Grado en Ingeniería Informática y Matemáticas  
Arquitectura de Computadores



Desconocido: 45338112 Guerrero Cano, Valentín



Inicio: Hoy, miércoles, 09:40:10

Final: Hoy, miércoles, 09:50:00

Preguntas: 10

Respuestas  
válidas:

Puntuación:

Nota:

**1**  
Elección única

En una máquina con 8 cores y tras ejecutar `export OMP_NUM_THREADS=4`, ¿cuántas iteraciones ejecuta la hebra máster en la región `parallel`?

```
int N = omp_get_max_threads();
omp_set_num_threads(2);
#pragma omp parallel for num_threads(6) if(N>=4) schedule(static)
for (int i=0 ; i <12 ; i++)
    printf(" thread: %d iteracion: %d \n", omp_get_thread_num(), i);
```

Usuario Profesores

- a) 4
- b) 12
- c) 6
- d) 2

**2**  
Elección única

El tiempo de ejecución de un programa paralelo ...

Usuario Profesores

- a) Siempre será menor que el de su versión secuencial, para cualquier tamaño del problema
- b) Puede ser mayor que el tiempo de la versión secuencial para tamaños de problema pequeños, debido a la sobrecarga introducida al crear y destruir las hebras
- c) Aumenta conforme el tamaño del problema disminuye
- d) Se reduce conforme el tamaño del problema aumenta


**3**  
Elección única

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

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

Usuario Profesores

- a) Ejecutar las ramas `if` y `else` del bloque estructurado en paralelo
- b) Las otras respuestas no son correctas
- c) No ejecutar el código del bloque estructurado si `n<=20`

-  d) Evitar la sobrecarga introducida al paralelizar el código para tamaños del problema pequeños





**4**  
Elección única

El siguiente código se ejecuta en paralelo sobre 2 hebras para  $N=1024$  repartiendo las iteraciones del bucle más externo usando la directiva *for* con la cláusula *schedule(static,chunk)*, se puede asegurar que todos los threads realizarán el mismo trabajo (es decir, el mismo número de operaciones).

```
int m[N][N], v[N], r[N] = {0};





for (int i = 0; i < N; ++i)
    for (int j = 0; j <= i; ++j)
        r[i] += m[i][j] * v[j];
```

Esta afirmación es:  
Usuario Profesores

-  a) En algunas ejecuciones será correcta y en otras no.
-  b) Correcta siempre en todas las ejecuciones.
-  c) Incorrecta en todas las ejecuciones.
-  d) Será correcta para algunos valores de chunk y para otros no.





**5**  
Elección única

Las variables de control internas de OpenMP  
Usuario Profesores

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





**6**  
Elección única

¿Cuál de las siguientes opciones permitiría comprobar qué tipo de planificación obtiene mejores resultados para un programa paralelo con ayuda de un *script*?  
Usuario Profesores

-  a) `schedule(runtime)`
-  b) `schedule(guided)`
-  c) `schedule(dynamic)`
-  d) `schedule(static)`


**7**  
Elección única




Si le piden que realice un estudio de escalabilidad de un código que calcula el producto de dos matrices.  
Usuario Profesores

-  a) No haría nada de lo indicado en el resto de respuestas.
-  b) Representaría en una gráfica el tiempo de ejecución en función del tamaño de las matrices.
-  c) Representaría en una gráfica la ganancia en velocidad (o ganancia en prestaciones) en función del número de núcleos.
-  d) Representaría en una gráfica el tiempo de ejecución en función del número de núcleos.

**8**  
Elección única

¿Cuál de las siguientes formas es la correcta para fijar a 4 el número de hebras para un programa OpenMP ?  
Usuario Profesores





-  a) En un programa OpenMP, usando la función `omp_max_threads(4)` al principio de la función main.

-  b) En un programa OpenMP, usando la función `omp_set_num_threads(4)` al principio de la función `main`.
-  c) En la consola del sistema, usando la variable de entorno `export OMP_THREAD_LIMIT=4`
-  d) En un programa OpenMP, usando la función `omp_num_threads(4)` al principio de la función `main`.

**9**

Elección única

¿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)` ?  
 Usuario Profesores

-  a) Usando la variable de entorno `OMP_SCHEDULE` **O** 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**


Elección única


¿Qué código cree mejor para conseguir multiplicar una matriz triangular *superior* por un vector?


```
int m[N][N], v[N], r[N] = {0};
```

Usuario Profesores

-  a) 

```
for (int i=0 ; i<N ; i++)
  for (int j=0 ; j<=i ; j++)
    r[i] += m[i][j] * v[j];
```
-  b) 

```
for (int i=0 ; i<N ; i++)
  for (int j=0 ; j<N ; j++)
    r[i] += m[i][j] * v[j];
```
-  c) 

```
for (int i=0 ; i<N ; i++)
  for (int j=i ; j<N ; j++)
    r[i] += m[i][j] * v[j];
```
-  d) 

```
for (int j=0 ; j<N ; j++)
  for (int i=0 ; i<N ; i++)
    r[i] += m[i][j] * v[j];
```