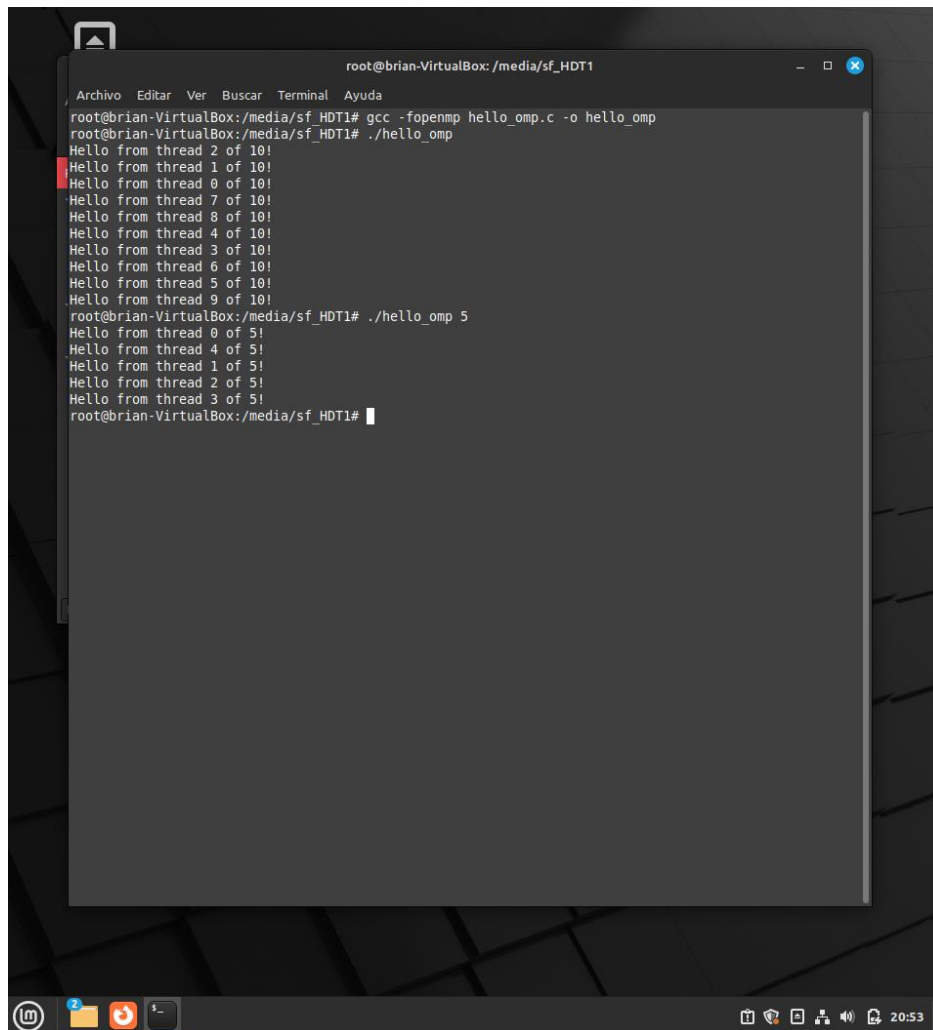


Contenido:

Ejercicio 1 (10 puntos)

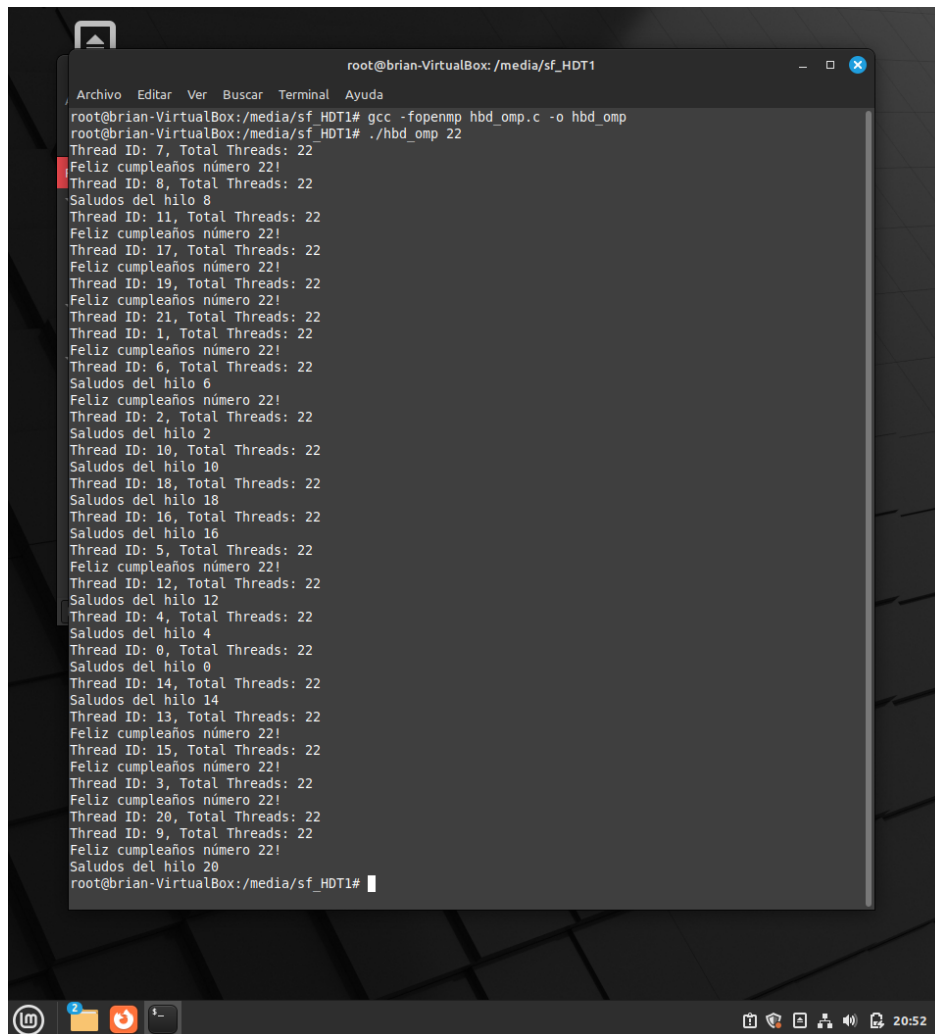


```
root@brian-VirtualBox: /media/sf_HDT1
Archivo Editar Ver Buscar Terminal Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc -fopenmp hello_omp.c -o hello_omp
root@brian-VirtualBox:/media/sf_HDT1# ./hello_omp
Hello from thread 2 of 10!
Hello from thread 1 of 10!
Hello from thread 0 of 10!
Hello from thread 7 of 10!
Hello from thread 8 of 10!
Hello from thread 4 of 10!
Hello from thread 3 of 10!
Hello from thread 6 of 10!
Hello from thread 5 of 10!
Hello from thread 9 of 10!
root@brian-VirtualBox:/media/sf_HDT1# ./hello_omp 5
Hello from thread 0 of 5!
Hello from thread 4 of 5!
Hello from thread 1 of 5!
Hello from thread 2 of 5!
Hello from thread 3 of 5!
root@brian-VirtualBox:/media/sf_HDT1#
```

PREGUNTA: ¿Por qué al ejecutar su código los mensajes no están desplegados en orden?

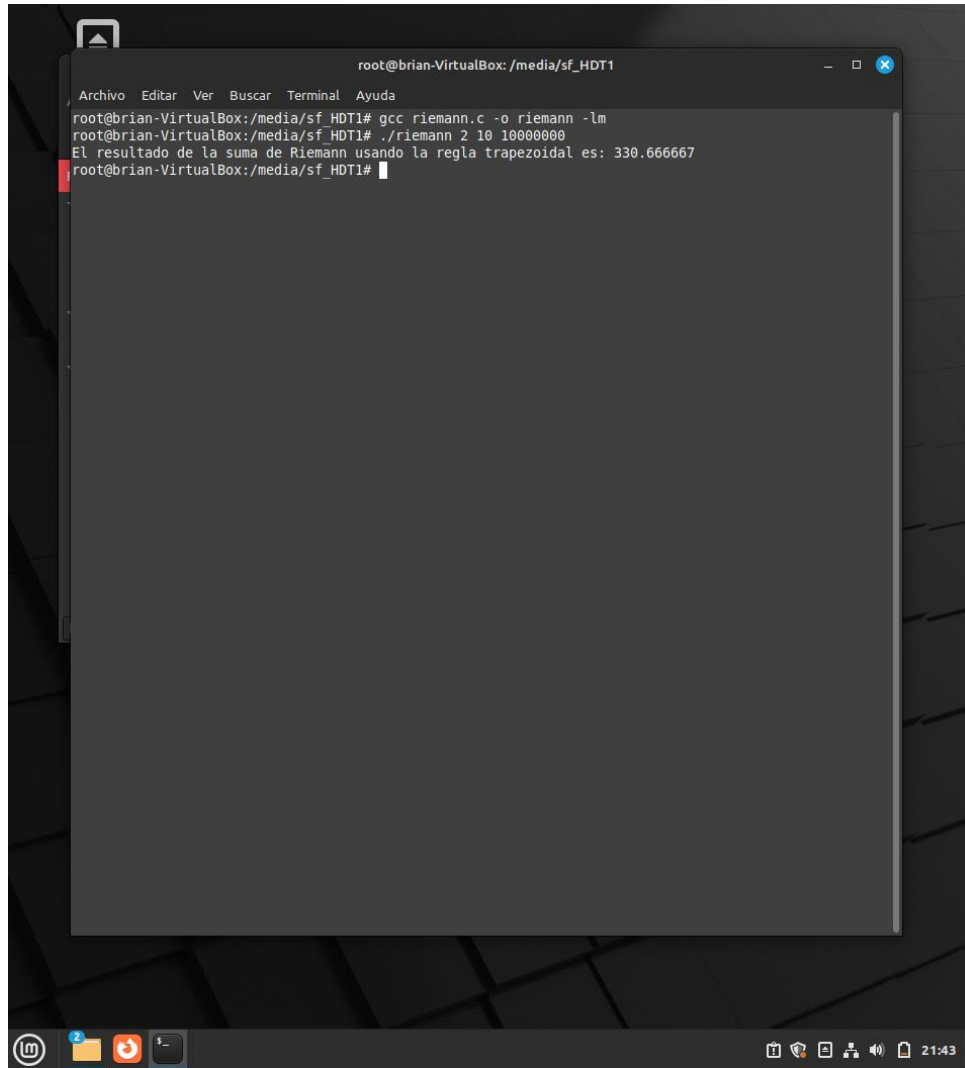
Porque los threads en OpenMP se ejecutan simultáneamente, e intentan utilizar printf. Esto se refleja en que el orden de impresión de la información no coincide con el orden numérico.

Ejercicio 2 (10 puntos)

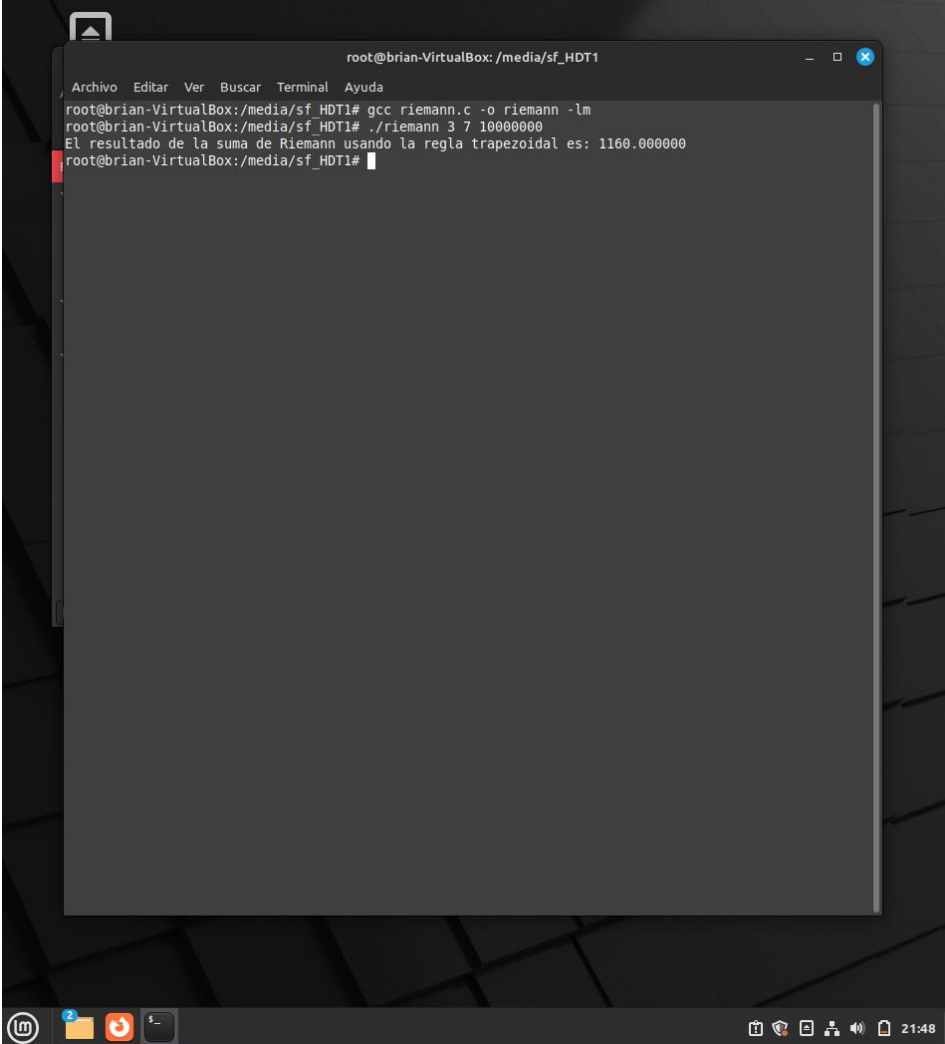


```
root@brian-VirtualBox: /media/sf_HDT1
Archivo Editar Ver Buscar Terminal Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc -fopenmp hbd_omp.c -o hbd_omp
root@brian-VirtualBox:/media/sf_HDT1# ./hbd_omp 22
Thread ID: 7, Total Threads: 22
Feliz cumpleaños número 22!
Thread ID: 8, Total Threads: 22
Saludos del hilo 8
Thread ID: 11, Total Threads: 22
Feliz cumpleaños número 22!
Thread ID: 17, Total Threads: 22
Feliz cumpleaños número 22!
Thread ID: 19, Total Threads: 22
Feliz cumpleaños número 22!
Thread ID: 21, Total Threads: 22
Thread ID: 1, Total Threads: 22
Feliz cumpleaños número 22!
Thread ID: 6, Total Threads: 22
Saludos del hilo 6
Feliz cumpleaños número 22!
Thread ID: 2, Total Threads: 22
Saludos del hilo 2
Thread ID: 10, Total Threads: 22
Saludos del hilo 10
Thread ID: 18, Total Threads: 22
Saludos del hilo 18
Thread ID: 16, Total Threads: 22
Saludos del hilo 16
Thread ID: 5, Total Threads: 22
Feliz cumpleaños número 22!
Thread ID: 12, Total Threads: 22
Saludos del hilo 12
Thread ID: 4, Total Threads: 22
Saludos del hilo 4
Thread ID: 0, Total Threads: 22
Saludos del hilo 0
Thread ID: 14, Total Threads: 22
Saludos del hilo 14
Thread ID: 13, Total Threads: 22
Feliz cumpleaños número 22!
Thread ID: 15, Total Threads: 22
Feliz cumpleaños número 22!
Thread ID: 3, Total Threads: 22
Feliz cumpleaños número 22!
Thread ID: 20, Total Threads: 22
Thread ID: 9, Total Threads: 22
Feliz cumpleaños número 22!
Saludos del hilo 20
root@brian-VirtualBox:/media/sf_HDT1#
```

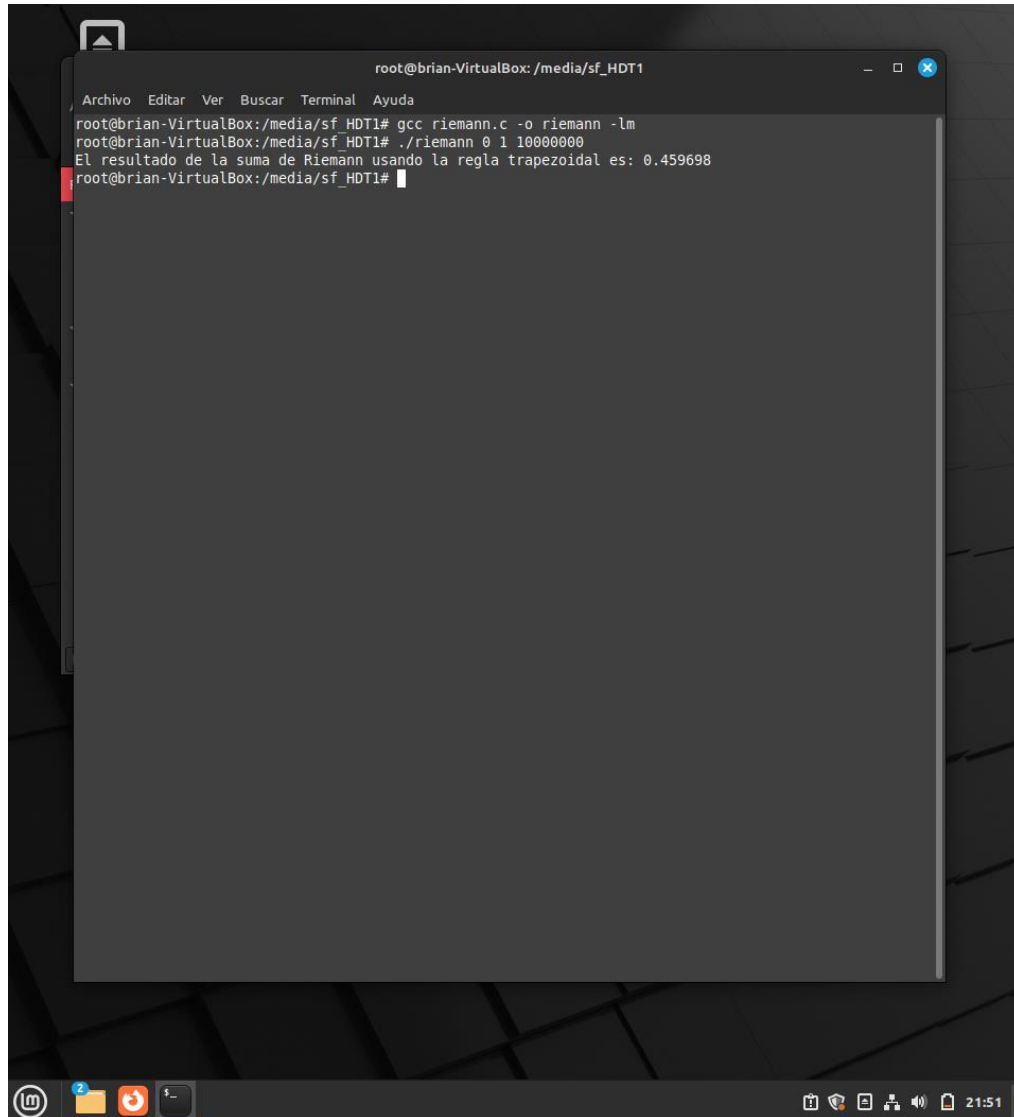
- x^2 (2, 10)



```
root@brian-VirtualBox: /media/sf_HDT1
Archivo Editar Ver Buscar Terminal Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc riemann.c -o riemann -lm
root@brian-VirtualBox:/media/sf_HDT1# ./riemann 2 10 10000000
El resultado de la suma de Riemann usando la regla trapezoidal es: 330.666667
root@brian-VirtualBox:/media/sf_HDT1#
```



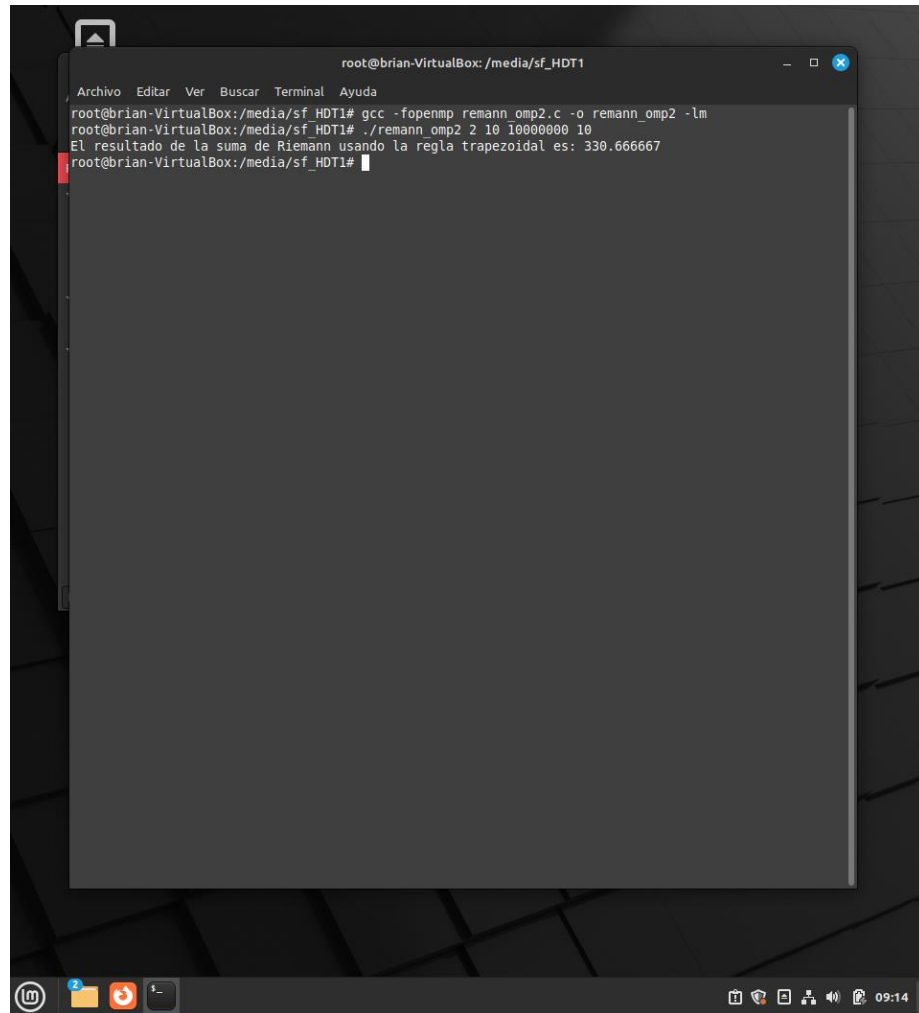
```
root@brian-VirtualBox: /media/sf_HDT1
Archivo Editar Ver Buscar Terminal Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc riemann.c -o riemann -lm
root@brian-VirtualBox:/media/sf_HDT1# ./riemann 3 7 10000000
El resultado de la suma de Riemann usando la regla trapezoidal es: 1160.000000
root@brian-VirtualBox:/media/sf_HDT1#
```



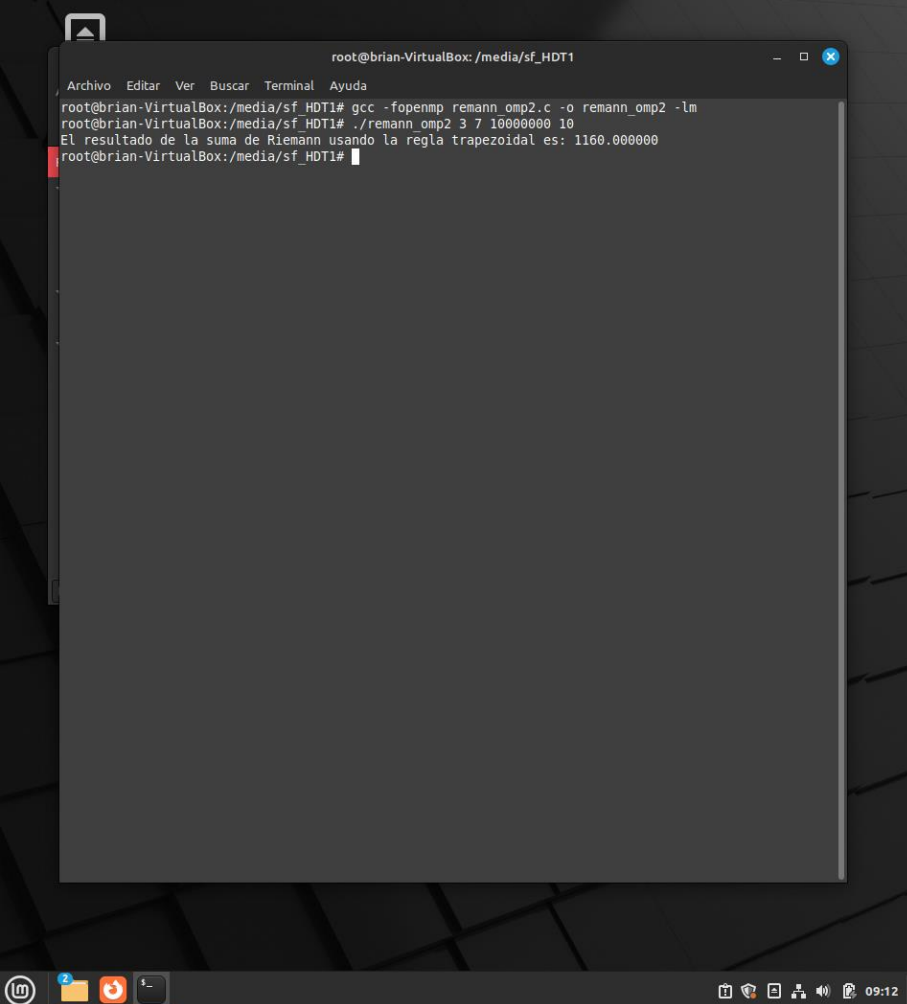
```
root@brian-VirtualBox: /media/sf_HDT1
Archivo Editar Ver Buscar Terminal Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc riemann.c -o riemann -lm
root@brian-VirtualBox:/media/sf_HDT1# ./riemann 0 1 10000000
El resultado de la suma de Riemann usando la regla trapezoidal es: 0.459698
root@brian-VirtualBox:/media/sf_HDT1#
```

Ejercicio 4 (20 puntos)

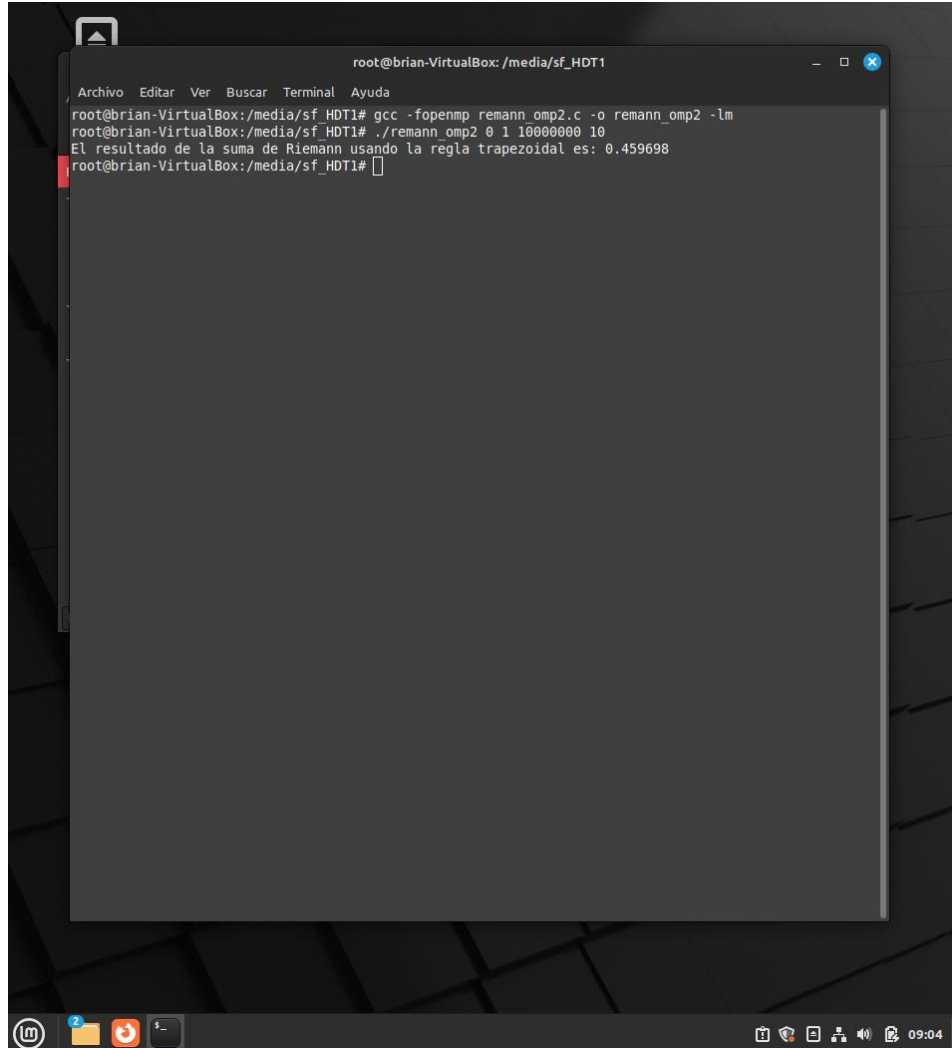
- x^2 (2, 10)



```
root@brian-VirtualBox: /media/sf_HDT1
Archivo Editar Ver Buscar Terminal Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc -fopenmp remann omp2.c -o remann_omp2 -lm
root@brian-VirtualBox:/media/sf_HDT1# ./remann_omp2 2 10 10000000 10
El resultado de la suma de Riemann usando la regla trapezoidal es: 330.666667
root@brian-VirtualBox:/media/sf_HDT1#
```



```
root@brian-VirtualBox: /media/sf_HDT1
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc -fopenmp remann_omp2.c -o remann_omp2 -lm
root@brian-VirtualBox:/media/sf_HDT1# ./remann_omp2 3 7 10000000 10
El resultado de la suma de Riemann usando la regla trapezoidal es: 1160.000000
root@brian-VirtualBox:/media/sf_HDT1#
```

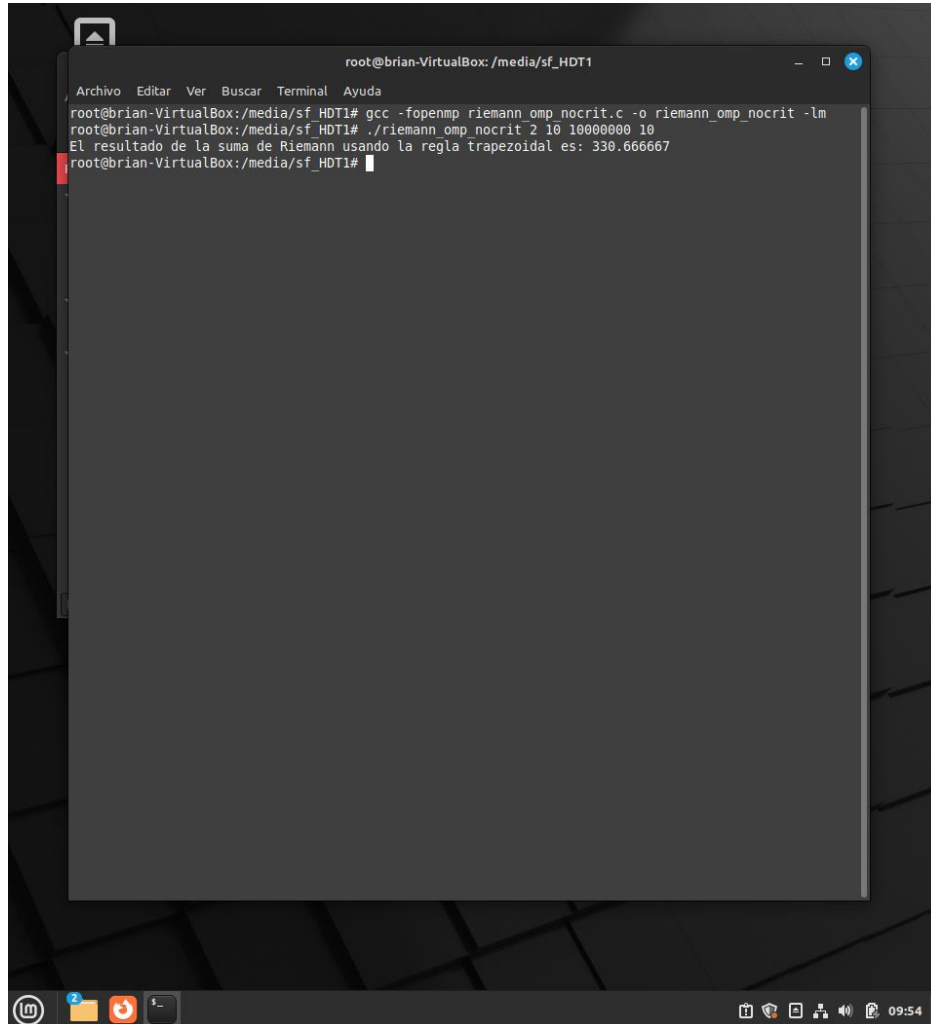


```
root@brian-VirtualBox: /media/sf_HDT1
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc -fopenmp remann omp2.c -o remann_omp2 -lm
root@brian-VirtualBox:/media/sf_HDT1# ./remann_omp2 0 1 10000000 10
El resultado de la suma de Riemann usando la regla trapezoidal es: 0.459698
root@brian-VirtualBox:/media/sf_HDT1#
```

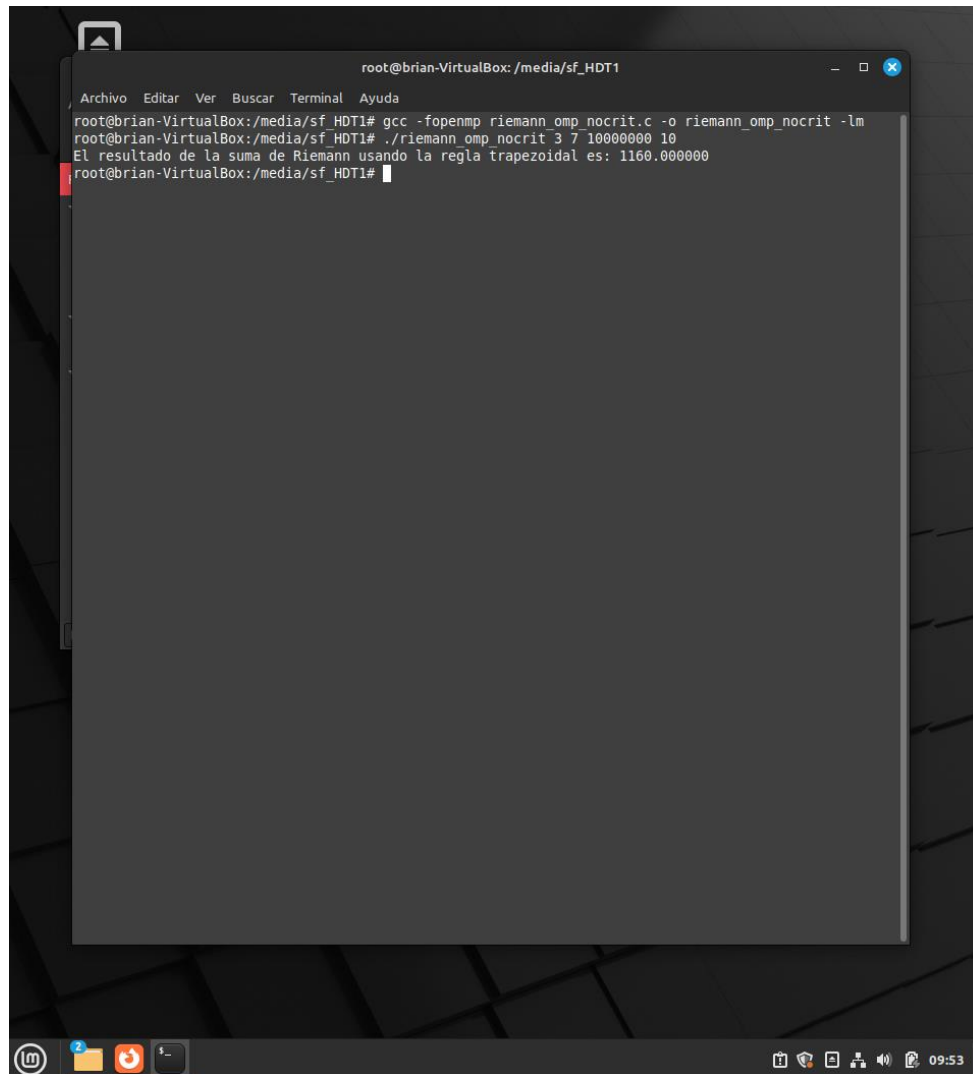
PREGUNTA: ¿Por qué es necesario el uso de la directiva `#pragma omp critical`?

Porque se necesita que la suma hacia la variable global de resultado sea realizada un thread a la vez, y así evitar condiciones de carrera.

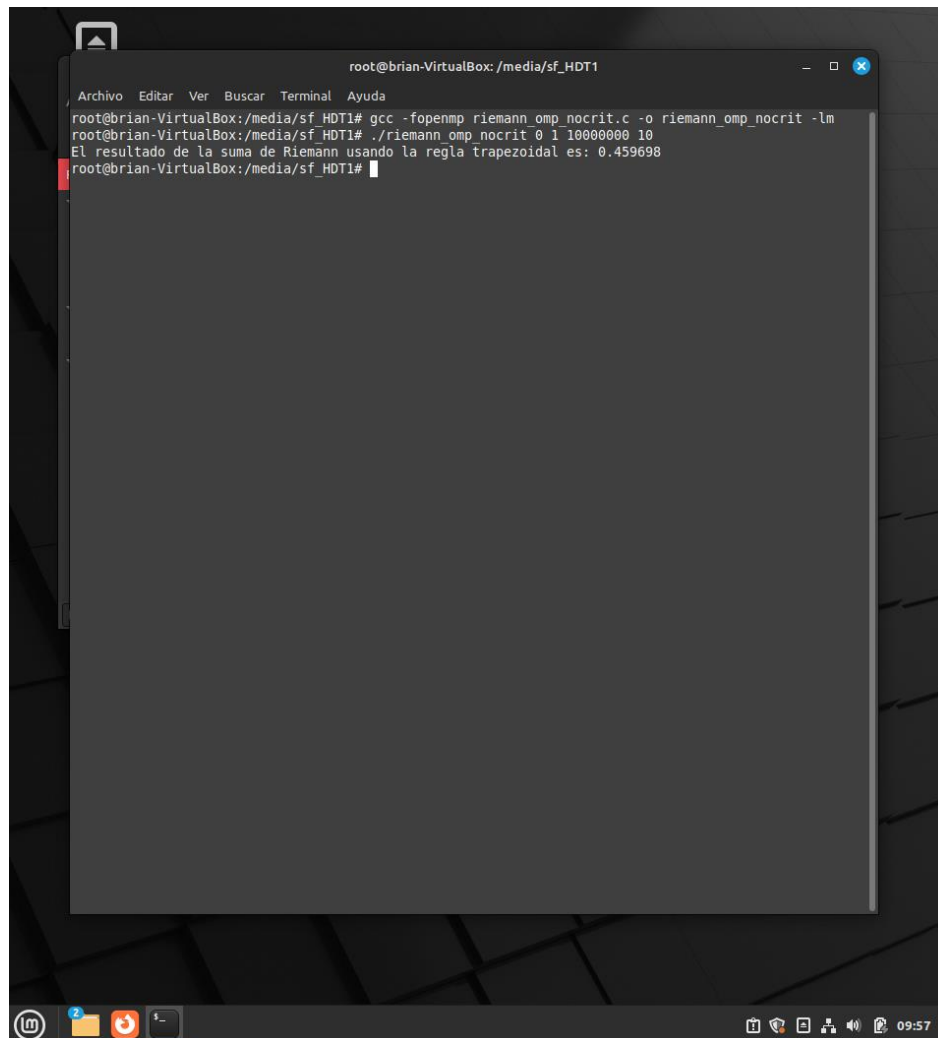
- x^2 (2, 10)



```
root@brian-VirtualBox: /media/sf_HDT1
Archivo Editar Ver Buscar Terminal Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc -fopenmp riemann_omp_nocrit.c -o riemann_omp_nocrit -lm
root@brian-VirtualBox:/media/sf_HDT1# ./riemann_omp_nocrit 2 10 10000000 10
El resultado de la suma de Riemann usando la regla trapezoidal es: 330.666667
root@brian-VirtualBox:/media/sf_HDT1#
```



```
root@brian-VirtualBox: /media/sf_HDT1
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc -fopenmp riemann_omp_nocrit.c -o riemann_omp_nocrit -lm
root@brian-VirtualBox:/media/sf_HDT1# ./riemann_omp_nocrit 3 7 10000000 10
El resultado de la suma de Riemann usando la regla trapezoidal es: 1160.000000
root@brian-VirtualBox:/media/sf_HDT1#
```



```
root@brian-VirtualBox: /media/sf_HDT1
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@brian-VirtualBox:/media/sf_HDT1# gcc -fopenmp riemann_omp_nocrit.c -o riemann_omp_nocrit -lm
root@brian-VirtualBox:/media/sf_HDT1# ./riemann_omp_nocrit 0 1 10000000 10
El resultado de la suma de Riemann usando la regla trapezoidal es: 0.459698
root@brian-VirtualBox:/media/sf_HDT1#
```

PREGUNTA: ¿Qué diferencia hay entre usar una variable global para añadir los resultados a un arreglo?

La diferencia radica en el acceso a la variable global. Al tener un arreglo con un tamaño preestablecido, cada thread modifica específicamente un espacio de memoria distinto, por lo que no existen condiciones de carrera. Por esta razón, no es necesario utilizar la directiva `#pragma omp critical`. Por otra parte, al utilizar un variable double que es modificada por cada thread, se está accediendo al mismo espacio en memoria, por lo que pueden ocurrir condiciones de carrera.

Referencias

Microsoft. (s.f.). Directivas de OpenMP. Obtenido de <https://learn.microsoft.com/es-es/cpp/parallel/openmp/reference/openmp-directives?view=msvc-170>