

2º curso / 2º cuatr.
Grado Ing. Inform.
Doble Grado Ing.
Inform. y Mat.

Arquitectura de Computadores (AC)

Cuaderno de prácticas.

Bloque Práctico 0. Entorno de programación

Estudiante (nombre y apellidos): Elena Cantero Molina

Grupo de prácticas: 2

Fecha de entrega:

Fecha evaluación en clase:

[-RECORDATORIO, quitar todo este texto en rojo del cuaderno definitivo–

1. COMENTARIOS

- 1) Esta **plantilla no sustituye al guion de prácticas, se ha preparado para ahorrarles trabajo**. Las preguntas de esta plantilla se han extraído del **guion** de prácticas de programación paralela, si tiene dudas sobre el enunciado consulte primero el **guion**.
- 2) Este cuaderno de prácticas se utilizará para asignarle una puntuación durante la evaluación continua de prácticas y también lo utilizará como material de estudio y repaso para preparar el examen de prácticas escrito. Luego redáctelo con cuidado, y sea ordenado y claro.
- 3) No use máquinas virtuales.

2. NORMAS SOBRE EL USO DE LA PLANTILLA

- 1) Usar **interlineado SENCILLO**.
 - 2) Respetar los tipos de letra y tamaños indicados:
 - Calibri-11 o Liberation Serif-11 para el texto
 - **Courier New-10 o Liberation Mono-10 para nombres de fichero, comandos, variables de entorno, etc., cuando se usan en el texto.**
 - **Courier New o Liberation Mono de tamaño 8 o 9 para el código fuente en los listados de código fuente.**
 - Formatee el código fuente de los listados para que sea legible, limpio y claro. Consulte, como ejemplo, los Listados 1 y 2 del guion (tabule, comente, ...)
 - 3) Insertar las capturas de pantalla donde se pidan y donde se considere oportuno
- Recuerde que debe adjuntar al zip de entrega, el pdf de este fichero, todos los ficheros con código fuente implementados/utilizados y el resto de ficheros que haya implementado/utilizado (scripts, hojas de cálculo, etc.), lea la Sección 1.4 del guion]**

Ejercicios basados en los ejemplos del seminario práctico

1. En el primer ejemplo de ejecución en atcgrid usando TORQUE se ejecuta el ejemplo HelloOMP.c usando la siguiente orden: `echo 'hello/HelloOMP' | qsub -q ac`. El resultado de la ejecución de este código en atcgrid se puede en el seminario. Conteste a las siguientes preguntas:
 - a. ¿Para qué se usa en qsub la opción -q?
RESPUESTA: Define el destino de un trabajo. El destino suele ser el nombre de una cola, un servidor o una cola dentro de un servidor.
 - b. ¿Cómo sabe el usuario que ha terminado la ejecución en atcgrid?
RESPUESTA: El usuario puede determinar que ha terminado la ejecución en atcgrid ejecutando el comando “qstat”, que muestra todos los trabajos que se están ejecutando.
 - c. ¿Cómo puede saber el usuario si ha habido algún error en la ejecución?
RESPUESTA: Ejecutando “qsub” se obtienen dos ficheros, uno STDIN.o que contiene el resultado y otro STDIN.e que contiene la lista de errores.
 - d. ¿Cómo ve el usuario el resultado de la ejecución?
RESPUESTA: Siguiendo lo dicho en la respuesta anterior, el usuario puede ver el resultado ejecutando en la ventana local “cat STDIN.o”.
 - e. ¿Por qué en el resultado de la ejecución aparecen 24 saludos “!!!Hello World!!!”?

Resto de ejercicios

4. Incorporar en el fichero .zip que se entregará al profesor el fichero /proc/cpuinfo de alguno de los nodos de atcgrid (atcgrid1, atcgrid2, atcgrid3), y del PC del aula de prácticas o de su PC. Indique qué ha hecho para obtener el contenido de /proc/cpuinfo en atcgrid.

RESPUESTA: Para obtener /proc/cpuinfo de mi PC he usado : “cat /proc/cpuinfo” y para los nodos de atcgrid: Primero he usado en la ssh la orden “echo 'cat /proc/cpuinfo' | qsub -q ac”, después en la sftp he ejecutado “get STDIN.o44738” y por último en la ventana local he ejecutado “cat STDIN.o44737”.

Teniendo en cuenta el contenido de cpuinfo conteste a las siguientes preguntas (justifique las respuestas):

a.¿Cuántos cores físicos y cuántos cores lógicos tiene el PC del aula de prácticas o su PC?

RESPUESTA: Mi PC tiene 4 cores lógicos y 2 cores físicos.

b.¿Cuántos cores físicos y cuántos cores lógicos tiene un nodo de atcgrid?

RESPUESTA: Tiene 24 cores lógicos y 12 cores físicos.

5. En el Listado 1 se puede ver un código fuente C que calcula la suma de dos vectores y en el Listado 2 una versión con C++:

```
v3 = v1 + v2; v3(i) = v1(i) + v2(i), i=0,...N-1
```

Los códigos utilizan directivas del compilador para fijar el tipo de variable de los vectores (v1, v2 y v3). En los comentarios que hay al principio de los códigos se indica cómo hay que compilarlos. Los vectores pueden ser:

- Variables locales: descomentando en el código #define VECTOR_LOCAL y comentando #define VECTOR_GLOBAL y #define VECTOR_DYNAMIC
- Variables globales: descomentando #define VECTOR_GLOBAL y comentando #define VECTOR_LOCAL y #define VECTOR_DYNAMIC
- Variables dinámicas: descomentando #define VECTOR_DYNAMIC y comentando #define VECTOR_LOCAL y #define VECTOR_GLOBAL. Si se usan los códigos tal y como están en Listado 1 y Listado 2, sin hacer ningún cambio, los vectores (v1, v2 y v3) serán variables dinámicas.

Por tanto, se debe definir sólo una de las siguientes constantes: VECTOR_LOCAL, VECTOR_GLOBAL o VECTOR_DYNAMIC.

a. En los dos códigos (Listado 1 y Listado 2) se utiliza la función clock_gettime() para obtener el tiempo de ejecución del trozo de código que calcula la suma de vectores. En el código se imprime la variable ncgt, ¿qué contiene esta variable? ¿qué información devuelve exactamente la función clock_gettime()? ¿en qué estructura de datos devuelve clock_gettime() la información (indicar el tipo de estructura de datos y describir la estructura de datos)?

RESPUESTA: ncgt contiene el tiempo transcurrido para sumar los vectores. Clock_gettime() nos devuelve la hora actual. El segundo parámetro de esta función es una estructura de tipo timespec que contiene al menos dos campos:

- tv_sec: número de segundos.

-tv_nsec: número de nanosegundos.

Restando estos dos valores se obtiene ncgt.

- b. Escribir en el cuaderno de prácticas las diferencias que hay entre el código fuente C y el código fuente C++ para la suma de vectores.

RESPUESTA:

Descripción diferencia	En C	En C++
Mostrar código	printf()	cout
Bibliotecas usadas	stdlib.h	cstdlib
	stdio.h	iostream
Reserva de memoria	malloc	new
Liberar memoria	free	delete

6. Generar el ejecutable del código fuente C del Listado 1 para vectores locales (para ello antes de compilar debe descomentar la definición de VECTOR_LOCAL y comentar las definiciones de VECTOR_GLOBAL y VECTOR_DYNAMIC). Ejecutar el código ejecutable resultante en atcgrid usando el la cola TORQUE. Incorporar volcados de pantalla que demuestren la ejecución correcta en atcgrid.

RESPUESTA:

```

E2estudiante7@atcgrid:~/sumavectores
[E2estudiante7@atcgrid sumavectores]$ rm *92
[E2estudiante7@atcgrid sumavectores]$ ls
sumavectores
[E2estudiante7@atcgrid sumavectores]$ echo 'sumavectores/sumavectores 3000' | qsub -q ac
45396.atcgrid
[E2estudiante7@atcgrid sumavectores]$ qstat
Job ID      Name      User              Time Use S Queue
-----
45396.atcgrid  STDIN    E2estudiante7    00:00:00 C ac
[E2estudiante7@atcgrid sumavectores]$ cat STDIN.e45396
[E2estudiante7@atcgrid sumavectores]$ cat STDIN.o45396
Tiempo(seg.):0.000015329 / Tamaño Vectores:3000 / V1[0]+V2[0]=V3[0](300.000000+300.000000=600.000000) / V1[2999]+V2[2999]=V3[2999](599.900000+0.100000=600.000000) /
[E2estudiante7@atcgrid sumavectores]$ clear

```

7. Ejecutar en atcgrid el código generado en el apartado anterior usando el script del Listado 3. Generar el ejecutable usando la opción de optimización -O2 tal y como se indica en el comentario que hay al principio del programa. Ejecutar el código también en su PC local para los mismos tamaños. ¿Se obtiene error para alguno de los tamaños? En caso afirmativo, ¿a qué se debe este error?

RESPUESTA: Ahora procedo a mostrar las líneas más relevantes del resultado que aparece en el fichero .o cuando ejecutamos el script en ATCGRID:

Nombre del trabajo especificado por usuario: SumaVectoresC_vlocales

Nodo que ejecuta qsub: atcgrid

Directorio en el que se ha ejecutado qsub: /home/E2estudiante7/sumavectores

Nodos asignados al trabajo: atcgrid1

Tiempo(seg.):0.000384330 / Tamaño Vectores:65536

Tiempo(seg.):0.000789889 / Tamaño Vectores:131072

Tiempo(seg.):0.001522148 / Tamaño Vectores:262144

A partir del siguiente tamaño aparecen los errores del tipo:

/var/lib/torque/mom_priv/jobs/45419.atcgrid.SC: line 20: 16840 Segmentation fault (core dumped) \$PBS_O_WORKDIR/sumavectores \$N

El problema es que los arrays están declarados como variables locales por lo que se almacenan en la pila. Y por lo tanto cuando se llegue a un valor muy grande del array se produzca violación del segmento.

A continuación muestro los resultados resumidos de la ejecución en mi ordenador PERSONAL:

Tiempo(seg.):0.000603291 / Tamaño Vectores:65536

Tiempo(seg.):0.001258130 / Tamaño Vectores:131072

Tiempo(seg.):0.002449811 / Tamaño Vectores:262144

Y a continuación se repite varias veces el siguiente error:

./Sumavectores.sh: línea 20: 18780 Violación de segmento (core generado) ./sumavectores \$N

Y el error se obtiene por los mismos motivos que el anterior.

8. Generar los ejecutables del código fuente C para vectores globales y para dinámicos. Genere el ejecutable usando -O2. Ejecutar los dos códigos en atcgrid usando un script como el del Listado 3 (hay que poner en el script el nombre de los ficheros ejecutables generados en este ejercicio) para el mismo rango de tamaños utilizado en el ejercicio anterior. Ejecutar también los códigos en su PC local. ¿Se obtiene error usando vectores globales o dinámicos? ¿A qué cree que es debido?

RESPUESTA: Ahora procedo a mostrar las líneas más relevantes del resultado que aparece en el fichero .o cuando ejecutamos el script en ATCGRID:

Nombre del trabajo especificado por usuario: SumaVectoresC_vglobales

Directorio en el que se ha ejecutado qsub: /home/E2estudiante7/sumavectores

Nodos asignados al trabajo: atcgrid1

Tiempo(seg.):0.000365304 / Tamaño Vectores:65536

Tiempo(seg.):0.000767750 / Tamaño Vectores:131072

Tiempo(seg.):0.001531613 / Tamaño Vectores:262144

Tiempo(seg.):0.003172201 / Tamaño Vectores:524288

Tiempo(seg.):0.005139789 / Tamaño Vectores:1048576

Tiempo(seg.):0.010108964 / Tamaño Vectores:2097152

Tiempo(seg.):0.019920362 / Tamaño Vectores:4194304

Tiempo(seg.):0.039878289 / Tamaño Vectores:8388608

Tiempo(seg.):0.079177964 / Tamaño Vectores:16777216

Tiempo(seg.):0.158224867 / Tamaño Vectores:33554432

Tiempo(seg.):0.172490141 / Tamaño Vectores:33554432

Nombre del trabajo especificado por usuario: SumaVectoresC_vdinamicos

Directorio en el que se ha ejecutado qsub: /home/E2estudiante7/sumavectores

Nodos asignados al trabajo: atcgrid1

Tiempo(seg.):0.000375086 / Tamaño Vectores:65536

Tiempo(seg.):0.000766601 / Tamaño Vectores:131072

Tiempo(seg.):0.001522244 / Tamaño Vectores:262144

Tiempo(seg.):0.003163102 / Tamaño Vectores:524288

Tiempo(seg.):0.004998804 / Tamaño Vectores:1048576

Tiempo(seg.):0.010017479	/ Tamaño Vectores:2097152
Tiempo(seg.):0.019925425	/ Tamaño Vectores:4194304
Tiempo(seg.):0.039817144	/ Tamaño Vectores:8388608
Tiempo(seg.):0.079276364	/ Tamaño Vectores:16777216
Tiempo(seg.):0.157102993	/ Tamaño Vectores:33554432
Tiempo(seg.):0.315638946	/ Tamaño Vectores:67108864

A continuación muestro los resultados resumidos de la ejecución en mi ordenador PERSONAL (primero con vectores globales y luego con vectores dinámicos):

Tiempo(seg.):0.000894471	/ Tamaño Vectores:65536
Tiempo(seg.):0.001727206	/ Tamaño Vectores:131072
Tiempo(seg.):0.004176379	/ Tamaño Vectores:262144
Tiempo(seg.):0.007281826	/ Tamaño Vectores:524288
Tiempo(seg.):0.009048598	/ Tamaño Vectores:1048576
Tiempo(seg.):0.020759719	/ Tamaño Vectores:2097152
Tiempo(seg.):0.038014738	/ Tamaño Vectores:4194304
Tiempo(seg.):0.064927385	/ Tamaño Vectores:8388608
Tiempo(seg.):0.130210613	/ Tamaño Vectores:16777216
Tiempo(seg.):0.225893816	/ Tamaño Vectores:33554432
Tiempo(seg.):0.203762226	/ Tamaño Vectores:33554432

Tiempo(seg.):0.000575225	/ Tamaño Vectores:65536
Tiempo(seg.):0.000444769	/ Tamaño Vectores:131072
Tiempo(seg.):0.000950686	/ Tamaño Vectores:262144
Tiempo(seg.):0.001738539	/ Tamaño Vectores:524288
Tiempo(seg.):0.003756257	/ Tamaño Vectores:1048576
Tiempo(seg.):0.007939759	/ Tamaño Vectores:2097152
Tiempo(seg.):0.018447514	/ Tamaño Vectores:4194304
Tiempo(seg.):0.050282940	/ Tamaño Vectores:8388608
Tiempo(seg.):0.099034809	/ Tamaño Vectores:16777216
Tiempo(seg.):0.239087541	/ Tamaño Vectores:33554432
Tiempo(seg.):0.481427207	/ Tamaño Vectores:67108864

Aquí no encontramos errores ni con vectores globales ni con vectores dinámicos. En el caso de las vectores globales la longitud de los arrays no está limitada por el tamaño de la pila del programa y en cuanto a las vectores dinámicos, el programa tiene memoria reutilizable durante la ejecución y los vectores pueden ser variables.

9. Rellenar una tabla como la Tabla 1 para atcgrid y otra para el PC local con los tiempos de ejecución obtenidos en los ejercicios anteriores para el trozo de código que realiza la suma de vectores. En la columna “Bytes de un vector” hay que poner el total de bytes reservado para un vector. Ayudándose de una hoja de cálculo represente en una misma gráfica los tiempos de ejecución obtenidos en atcgrid para vectores locales, globales y dinámicos (eje y) en función del tamaño en bytes de un vector (eje x). Utilice escala logarítmica en el eje de ordenadas (eje y) en todas las gráficas. ¿Hay diferencias en los tiempos de ejecución con vectores locales, globales y dinámicos?

RESPUESTA:

Tabla 1 . Tiempo de ejecución de la suma de vectores en Atcgrid

Nº de Componentes	Bytes de un vector	Tiempo para vect. locales	Tiempo para vect. globales	Tiempo para vect. dinámicos
65536	524288	0.000384330	0.000365304	0.000375086
131072	1048576	0.000789889	0.000767750	0.000766601
262144	2097152	0.001522148	0.001531613	0.001522244
524288	4194304	-	0.003172201	0.003163102
1048576	8388608	-	0.005139789	0.004998804
2097152	16777216	-	0.010108964	0.010017479
4194304	33554432	-	0.019920362	0.019925425
8388608	67108864	-	0.039878289	0.039817144
16777216	134217728	-	0.079177964	0.079276364
33554432	268435456	-	0.158224867	0.157102993
67108864	536870912	-	0.172490141	0.315638946

Tabla 2 . Tiempo de ejecución de la suma de vectores en Mi PC

Nº de Componentes	Bytes de un vector	Tiempo para vect. locales	Tiempo para vect. globales	Tiempo para vect. dinámicos
65536	524288	0.000603291	0.000894471	0.000575225
131072	1048576	0.001258130	0.001727206	0.000444769
262144	2097152	0.002449811	0.004176379	0.000950686
524288	4194304	-	0.007281826	0.001738539
1048576	8388608	-	0.009048598	0.003756257
2097152	16777216	-	0.020759719	0.007939759
4194304	33554432	-	0.038014738	0.018447514
8388608	67108864	-	0.064927385	0.050282940
16777216	134217728	-	0.130210613	0.099034809
33554432	268435456	-	0.225893816	0.239087541
67108864	536870912	-	0.203762226	0.481427207

Tabla 3 . Gráfica de tiempos de ejecución para los distintos tamaños en ATCGRID

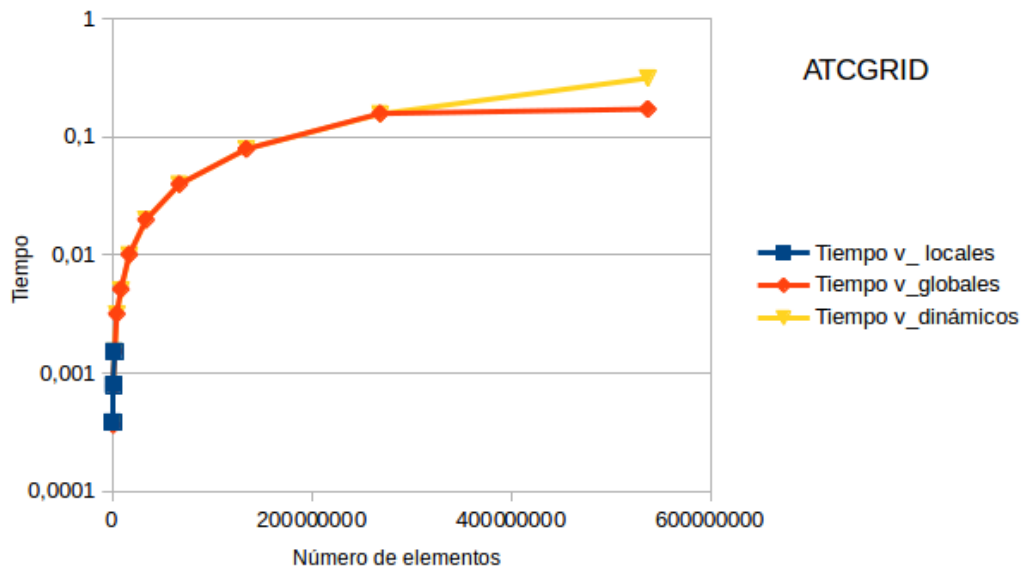
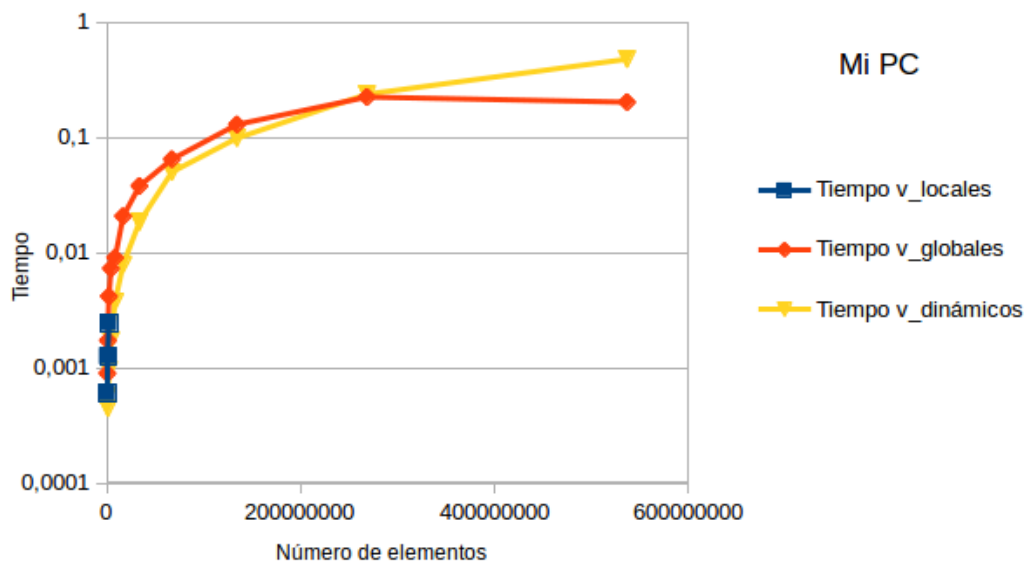


Tabla 4 . Gráfica de tiempos de ejecución para los distintos tamaños en Mi PC



Las diferencias entre vectores locales y globales o dinámicos son muy difíciles de determinar gráficamente. Para valores altos podemos observar que los vectores globales se comportan mejor que los dinámicos y esto podría explicar el caso en el que con los dinámicos hace falta reservar y liberar memoria utilizada por los vectores

10. Modificar el código fuente C para que el límite de los vectores cuando se declaran como variables globales sea igual al máximo número que se puede almacenar en la variable N ($MAX=2^{32}-1$). Generar el ejecutable usando variables globales. ¿Qué ocurre? ¿A qué es debido? Razone además por qué el máximo número que se puede almacenar en N es $2^{32}-1$.

RESPUESTA: Al intentar crear el ejecutable de Sumavectores.c obtenemos lo siguiente en la terminal:

/tmp/ccTOoQHy.o: En la función `main':

Sumavectores.c:(.text.startup+0x79): reubicación truncada para ajustar: R_X86_64_32S contra el

símbolo ``v2'` definido en la sección COMMON en `/tmp/ccTOoQHy.o`

Esto es debido a que nos pasamos del tamaño y el compilador detecta que hay un desbordamiento, el máximo número a almacenar en N es $2^{32}-1$ porque es el máximo número que se puede representar en un entero de 32bits y se le resta 1 a 2^{32} porque hay que tener en cuenta el 0.