

Arquitectura de Computadores (AC)

Cuaderno de prácticas.

Bloque Práctico 0. Entorno de programación

Estudiante (nombre y apellidos): Arturo Cortés Sánchez

Grupo de prácticas y profesor de prácticas: C2

Fecha de entrega:

Fecha evaluación en clase:

Antes de comenzar a realizar el trabajo de este cuaderno consultar el fichero con los normas de prácticas que se encuentra en SWAD

Parte I. Ejercicios basados en los ejemplos del seminario práctico

Crear el directorio con nombre bp0 en atcgrid y en el PC local.

1. Ejecutar lscpu en el PC y en un nodo de cómputo de atcgrid.

(a) Mostrar con capturas de pantalla el resultado de estas ejecuciones.

RESPUESTA:

ATCGRID:

The image displays three terminal windows side-by-side, illustrating the execution of the `lscpu` command. The leftmost window shows a terminal session on the `C2estudiante3@atcgrid` node, where the user `ArturoCortésSánchez` runs `lscpu` and the output shows system information for the `STDIN.o1944` node. The middle window shows a terminal session on the local PC `arturo@mior-pc`, where the user runs `sftp C2estudiante3@atcgrid.ugr.es` to connect to the remote node. The rightmost window shows the output of the `lscpu` command on the local PC, displaying detailed hardware specifications for the Intel(R) Xeon(R) CPU E5645 @ 2.40GHz, including architecture, cache sizes, and virtualization support.

```
C2estudiante3@atcgrid:~ 48x38
[ArturoCortésSánchez C2estudiante3@atcgrid:~] 20
19-02-20 miércoles
$echo 'lscpu' | qsub -q ac
1944.atcgrid
[ArturoCortésSánchez C2estudiante3@atcgrid:~] 20
19-02-20 miércoles
$ls -lag
total 64
drwx----- 7 C2estudiante3 4096 feb 20 16:35
.
drwxr-xr-x. 573 root 20480 feb 19 13:33
..
-rw----- 1 C2estudiante3 681 feb 20 16:30
.bash_history
-rw-r--r-- 1 C2estudiante3 18 ene 16 2015
.bash_logout
-rw-r--r-- 1 C2estudiante3 193 ene 16 2015
.bash_profile
-rw-r--r-- 1 C2estudiante3 231 ene 16 2015
.bashrc
drwxrwxr-x 3 C2estudiante3 4096 feb 20 15:52
.cache
drwxrwxr-x 3 C2estudiante3 4096 feb 20 15:52
.config
drwxr-xr-x 3 C2estudiante3 4096 feb 26 2016
.local
drwxr-xr-x 4 C2estudiante3 4096 ene 30 2015
.mozilla
drwxr-xr-x 2 C2estudiante3 4096 ene 21 12:45
.ssh
-rw----- 1 C2estudiante3 0 feb 20 16:31
STDIN.e1944
-rw----- 1 C2estudiante3 1253 feb 20 16:31
STDIN.o1944
[ArturoCortésSánchez C2estudiante3@atcgrid:~] 20
19-02-20 miércoles
$
```

```
arturo@mior-pc:~
[arturo@mior-pc ~]$ sftp C2estudiante3@atcgrid.ugr.es
r.es
C2estudiante3@atcgrid.ugr.es's password:
Connected to C2estudiante3@atcgrid.ugr.es.
sftp> ls
STDIN.e1944 STDIN.o1944
sftp> get ST
STDIN.e1944 STDIN.o1944
sftp> get STDIN.*
STDIN.e1944 STDIN.o1944
sftp> get STDIN.o1944
Fetching /home/C2estudiante3/STDIN.o1944 to STDIN
.o1944
/home/C2estudi 100% 1253 85.3KB/s 00:00
sftp>
```

```
arturo@mior-pc:~ 49x38
$cat STDIN.o1944
Architecture: x86_64
CPU op-mode(s): 32-bit, 64-bit
Byte Order: Little Endian
CPU(s): 24
On-line CPU(s) list: 0-23
Thread(s) per core: 2
Core(s) per socket: 6
Socket(s): 2
NUMA node(s): 2
Vendor ID: GenuineIntel
CPU family: 6
Model: 44
Model name: Intel(R) Xeon(R) CPU
E5645 @ 2.40GHz
Stepping: 2
CPU MHz: 2401.000
CPU max MHz: 2401.0000
CPU min MHz: 1600.0000
BogoMIPS: 4800.24
Virtualization: VT-x
L1d cache: 32K
L1i cache: 32K
L2 cache: 256K
L3 cache: 12288K
NUMA node0 CPU(s): 0-5,12-17
NUMA node1 CPU(s): 6-11,18-23
Flags: fpu vme de pse tsc msr pae
mce cx8 apic sep mtrr pge mca cmov pat pse36 clf
lush dts acpi mmx fxsr sse sse2 ss ht tm pbe sysc
all nx pdpe1gb rdtscp lm constant_tsc arch_perfm
n pebs bts rep good nopl xtopology nonstop_tsc ap
erfmperf eagerfpu pni dtes64 monitor ds_cpl vmx s
mx est tm2 ssse3 cx16 xtpr pdcm pcid dca sse4_1 s
se4_2 popcnt lahf_lm epb ssbd ibrs ibpb stibp tpr
shadow vmx flexpriority ept vpid dtherm ida ara
t spec_ctrl intel stibp flush lld
[ArturoCortésSánchez arturo@mior-pc:~] 2019-02-20
miércoles
```

PC:

```

arturo@mior-pc:~
arturo@mior-pc:~ 80x36
[ArturoCortésSánchez arturo@mior-pc:~] 2019-02-20 miércoles
$lscpu
Arquitectura:                x86_64
modo(s) de operación de las CPUs:  32-bit, 64-bit
Orden de los bytes:          Little Endian
Tamaños de las direcciones:    36 bits physical, 48 bits virtual
CPU(s):                      4
Lista de la(s) CPU(s) en línea:  0-3
Hilo(s) de procesamiento por núcleo: 2
Núcleo(s) por «socket»:        2
«Socket(s)»                   1
Modo(s) NUMA:                  1
ID de fabricante:              GenuineIntel
Familia de CPU:                 6
Modelo:                         37
Nombre del modelo:              Intel(R) Core(TM) i3 CPU          M 380  @ 2.5
3GHz
Revisión:                       5
CPU MHz:                        1142.285
CPU MHz máx.:                   2533,0000
CPU MHz mín.:                    933,0000
BogoMIPS:                       5055.23
Virtualización:                 VT-x
Caché L1d:                       32K
Caché L1i:                       32K
Caché L2:                        256K
Caché L3:                        3072K
CPU(s) del nodo NUMA 0:          0-3
Indicadores:                     fpu vme de pse tsc msr pae mce cx8 apic sep
mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe sys
call nx rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology non
stop_tsc cpuid aperfmperf pni dtes64 monitor ds_cpl vmx est tm2 ssse3 cx16 xtpr
pdcm pcid sse4_1 sse4_2 popcnt lahf_lm pti ssbd ibrs ibpb stibp tpr_shadow vnmi
flexpriority ept vpid dtherm arat flush_lld
[ArturoCortésSánchez arturo@mior-pc:~] 2019-02-20 miércoles
$

```

(b) ¿Cuántos cores físicos y cuántos cores lógicos tienen los nodos de cómputo de atcgrid y del PC? Razonar las respuestas

RESPUESTA:

En las siguientes líneas de lscpu podemos ver el número de núcleos así como el número de hilos por núcleo

ATCGRID:

CPU(s): 24

On-line CPU(s) list: 0-23

Thread(s) per core: 2

PC:

CPU(s): 4

Lista de la(s) CPU(s) en línea: 0-3

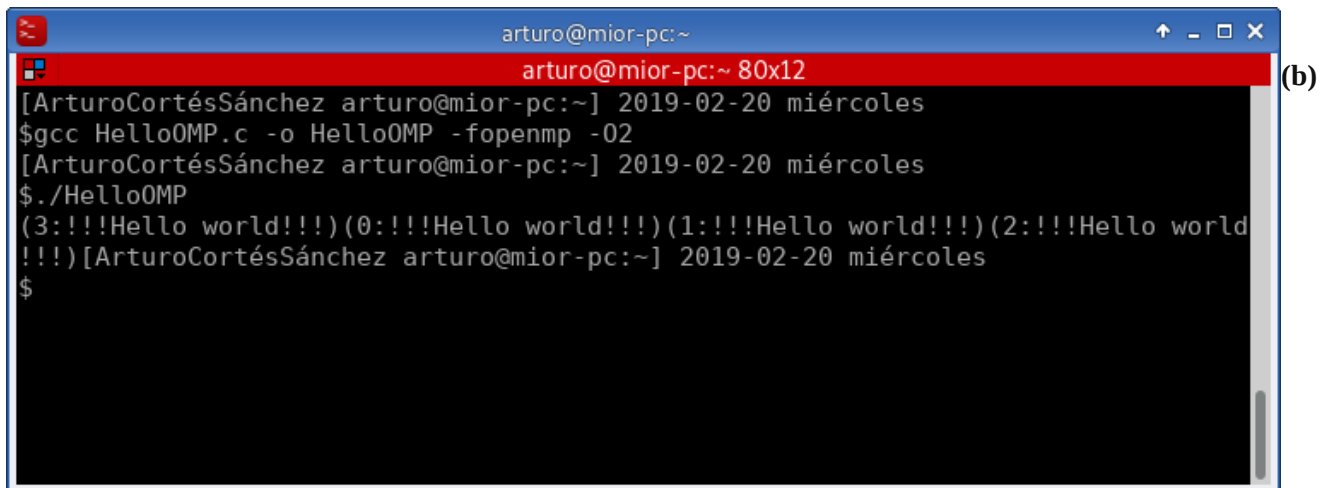
Hilo(s) de procesamiento por núcleo: 2

Sabiendo esto podemos calcular que cada nodo de atcgrid tiene 12 núcleos físicos y 24 lógicos y que mi PC tiene 2 núcleos físicos y 4 lógicos.

2. Compilar y ejecutar en el PC el código HelloOMP.c del seminario (recordar que se debe usar un directorio independiente para cada ejercicio dentro de bp0 que contenga todo lo utilizado, implementado o generado durante el desarrollo del mismo, para el presente ejercicio el directorio sería ejer2, como se indica en las normas de prácticas).

(a) Adjuntar capturas de pantalla que muestren la compilación y ejecución en el PC.

RESPUESTA:

 (b)

Justificar el número de “Hello world” que se imprimen en pantalla en ambos casos teniendo en cuenta la salida que devuelve lscpu.

RESPUESTA:

En la siguiente línea de lscpu podemos ver que mi pc tiene cuatro núcleos

CPU(s): 4

Por lo que sale un Hello world por cada núcleo.

3. Copiar el ejecutable de HelloOMP.c que ha generado anteriormente y que se encuentra en el directorio ejer2 del PC al directorio ejer2 de su home en el front-end de atcgrid. Ejecutar (desde el directorio de este ejercicio, ejer3) este código en un nodo de cómputo de atcgrid usando la cola ac del gestor de colas (no use ningún script).

(a) Adjuntar capturas de pantalla que muestren la copia del fichero, el envío a la cola de la ejecución y el resultado de esta ejecución tal y como la devuelve el gestor de colas.

RESPUESTA:

The image shows three terminal windows. The left window shows the output of 'ls -lag' for a directory containing 'HelloOMP.c'. The middle window shows an 'sftp' session where 'HelloOMP.c' is uploaded to a remote directory and then executed. The right window shows the output of the execution, displaying a list of 'Hello world' messages with thread IDs.

(b) Justificar el número de “Hello world” que se observan en el resultado teniendo en cuenta la salida que devuelve lscpu.

RESPUESTA:

Como hemos visto en el lscpu cada nodo de atcgrid tiene 24 núcleos, por tanto tenemos 24 Hello worlds

- Modificar en su PC HelloOMP.c para que se imprima “world” en un printf distinto al usado para “Hello”, en ambos printf se debe imprimir el identificador del thread que escribe en pantalla. Nombrar al código resultante HelloOMP2.c. Compilar este nuevo código en el PC y ejecutarlo. Copiar el fichero ejecutable resultante en el front-end de atcgrid (directorio ejer4). Ejecutar el código en un nodo de cómputo de atcgrid usando el script script_helloomp.sh del seminario (el nombre del ejecutable en el script debe ser HelloOMP2).

(a) Adjuntar capturas de pantalla que muestren el nuevo código, la compilación, la copia a atcgrid, el envío a la cola de la ejecución y el resultado de esta ejecución tal y como la devuelve el gestor de colas.

RESPUESTA:

The image shows a terminal window where 'HelloOMP2.c' is compiled with 'gcc' and then executed. The output shows 'Hello' and 'world' messages with thread IDs.

The image shows three terminal windows. The left window shows a local user running 'ls -lag' and 'sftp' commands to connect to a remote node. The middle window shows the user navigating the remote file system, listing files, and uploading a script. The right window shows the output of the script execution, displaying thread counts and 'Hello' messages from different threads.

```

C2estudiante3@atcgrid:~/bp0/ejer4 61x46
[ArturoCortésSánchez C2estudiante3@atcgrid:~/bp0/ejer4] 2019-02-20 miércoles
$ sftp -l arturo@atcgrid
sftp> ls
Remote working directory: /home/C2estudiante3/bp0
sftp> cd ejer4
sftp> ls
sftp> lpwd
Local working directory: /home/arturo
sftp> lcd bp0
sftp> lls
ejer2/ ejer3/ ejer4/
sftp> lcd ejer4/
sftp> lls
HelloOMP2 HelloOMP2.c script_helloomp.sh
sftp> pwd
Remote working directory: /home/C2estudiante3/bp0/ejer4
sftp> cd ejer4
sftp> ls
HelloOMP2 HelloOMP2.c script_helloomp.sh
sftp> put H
HelloOMP2 HelloOMP2.c
sftp> put HelloOMP2
Uploading HelloOMP2 to /home/C2estudiante3/bp0/ejer4/HelloOMP2
HelloOMP2 100% 16KB 325.5KB/s 00:00
sftp> ls
HelloOMP2
sftp> put script_helloomp.sh
Uploading script_helloomp.sh to /home/C2estudiante3/bp0/ejer4/
script_helloomp.sh 100% 826 4.0KB/s 00:00
sftp> get he
helloomp.e2029 helloomp.o2029
sftp> get helloomp.o2029
Fetching /home/C2estudiante3/bp0/ejer4/helloomp.o2029 to helloomp.o2029
/home/C2estudi 100% 507 35.8KB/s 00:00
sftp>

arturo@mior-pc:~/bp0/ejer4 63x46
$ cat helloomp.o2029
Id. usuario del trabajo: C2estudiante3
Id. del trabajo: 2029.atcgrid
Nombre del trabajo especificado por usuario: helloomp
Modo que ejecuta qsub: atcgrid
Cola: ac
Nodos asignados al trabajo:
atcgrid2
Nº de threads inicial: 12

Para 12 threads:
Hello 3
Hello 11
Hello 10
Hello 6
Hello 2
Hello 1
Hello 0
Hello 7
Hello 9
Hello 4
Hello 5
Hello 8
world 0

Para 6 threads:
Hello 1
Hello 0
Hello 2
Hello 5
Hello 3
Hello 4
world 0

Para 3 threads:
Hello 2
Hello 1
Hello 0
world 0

Para 1 threads:
Hello 0
world 0
[ArturoCortésSánchez arturo@mior-pc:~/bp0/ejer4] 2019-02-20 miércoles

```

The image shows a code editor window titled 'HelloOMP2.c - Visual...'. It displays the source code of a script named 'script_helloomp.sh'. The code includes headers for OpenMP and stdio, and contains a main function that prints 'Hello' and 'world' messages using OpenMP thread IDs.

```

1 #include <omp.h>
2 #include <stdio.h>
3 int main(void) {
4     #pragma omp parallel
5     printf("Hello %d\n", omp_get_thread_num());
6     printf("world %d\n", omp_get_thread_num());
7     return (0);
8 }

```

(b) ¿Qué nodo de cómputo de atcgrid ha ejecutado el script? Explicar cómo ha obtenido esta información.

RESPUESTA:

Lo ha ejecutado en atcgrid2, lo cual se puede ver en la línea que dice “nodos asignados al trabajo”

(c) ¿Qué ocurre si se ejecuta el script usando ./HelloOMP2 en lugar de \$PBS_O_WORKDIR/HelloOMP2? Razonar respuesta y adjuntar capturas de pantalla que muestren lo que ocurre.

RESPUESTA:

Que el archivo con la salida contiene lo siguiente:

Id. usuario del trabajo: C2estudiante3

Id. del trabajo: 2038.atcgrid

Nombre del trabajo especificado por usuario: helloomp

Nodo que ejecuta qsub: atcgrid

Cola: ac

Depto. Arquitectura y Tecnología de Computadores

Nodos asignados al trabajo:

atcgrid2

Nº de threads inicial: 12

Para 12 threads:

Para 6 threads:

Para 3 threads:

Para 1 threads:

Y el archivo de error:

/var/spool/pbs/mom_priv/jobs/2038.atcgrid.SC: línea 22: ./HelloOMP2: No existe el fichero o el directorio

/var/spool/pbs/mom_priv/jobs/2038.atcgrid.SC: línea 22: ./HelloOMP2: No existe el fichero o el directorio

/var/spool/pbs/mom_priv/jobs/2038.atcgrid.SC: línea 22: ./HelloOMP2: No existe el fichero o el directorio

/var/spool/pbs/mom_priv/jobs/2038.atcgrid.SC: línea 22: ./HelloOMP2: No existe el fichero o el directorio

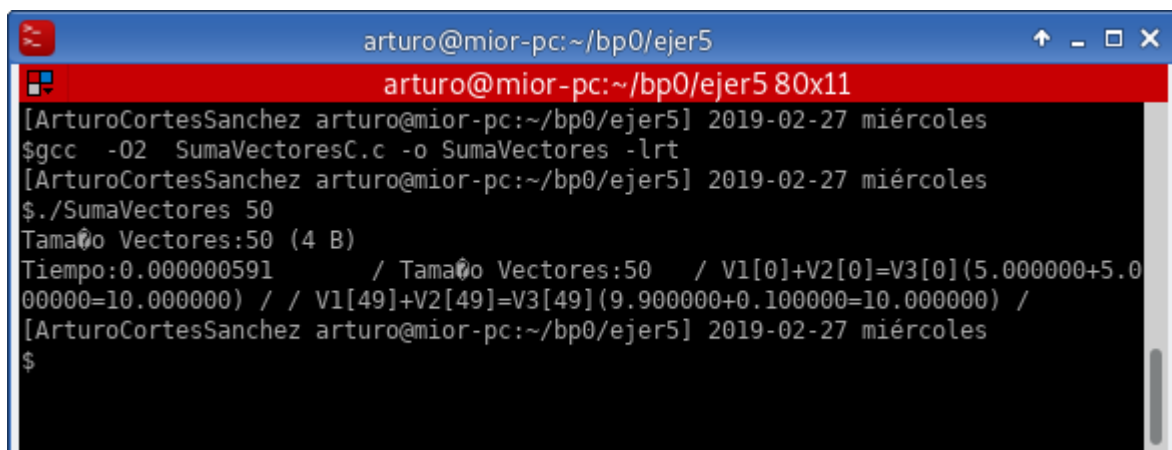
Viendo esto podemos deducir que hacer los cambios indicados en el ejercicio causa que no encuentre el ejecutable.

Parte II. Resto de ejercicios

RESPUESTA:

5. Generar en el PC 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). El comentario inicial del código muestra la orden para compilar (siempre hay que usar -O2 al compilar como se indica en las normas de prácticas). Incorporar volcados de pantalla que demuestren la compilación y la ejecución correcta del código en el PC (leer lo indicado al respecto en las normas de prácticas).

RESPUESTA:



```
arturo@mior-pc: ~/bp0/ejer5
arturo@mior-pc:~/bp0/ejer5 80x11
[ArturoCortesSanchez arturo@mior-pc:~/bp0/ejer5] 2019-02-27 miércoles
$gcc -O2 SumaVectoresC.c -o SumaVectores -lrt
[ArturoCortesSanchez arturo@mior-pc:~/bp0/ejer5] 2019-02-27 miércoles
$./SumaVectores 50
Tamaño Vectores:50 (4 B)
Tiempo:0.000000591 / Tamaño Vectores:50 / V1[0]+V2[0]=V3[0] (5.000000+5.000000=10.000000) / / V1[49]+V2[49]=V3[49] (9.900000+0.100000=10.000000) /
[ArturoCortesSanchez arturo@mior-pc:~/bp0/ejer5] 2019-02-27 miércoles
$
```

6. En el código del Listado 1 se utiliza la función `clock_gettime()` para obtener el tiempo de ejecución del trozo de código que calcula la suma de vectores. El código se imprime la variable `ncgt`,

(a) ¿qué contiene esta variable?

RESPUESTA:

ncgt contiene el tiempo en nanosegundos transcurrido entre la primera y la segunda llamada a `clock_gettime()`

(b) ¿en qué estructura de datos devuelve `clock_gettime()` la información de tiempo (indicar el tipo de estructura de datos, describir la estructura de datos, e indicar los tipos de datos que usa)?

RESPUESTA:

Devuelve un struct llamado `timespect` que contiene dos campos, el primero de tipo `time_t`, llamado `tv_sec` y el segundo de tipo `long`, llamado `tv_nsec`.

```
struct timespec {
    time_t    tv_sec;        /* seconds */
    long      tv_nsec;       /* nanoseconds */
};
```

(c) ¿qué información devuelve exactamente la función `clock_gettime()` en la estructura de datos descrita en el apartado (b)? ¿qué representan los valores numéricos que devuelve?

RESPUESTA:

El campo `tv_sec` contiene el numero de segundos transcurridos desde el 1 de enero de 1970, y el campo `tv_nsec` contiene los nanosegundos transcurridos en el segundo actual.

7. Ejecutar en atcgrid el código generado en el apartado anterior usando el script del Listado 2. Ejecutar el código también en el PC para los mismos tamaños de los vectores. ¿Se obtiene error para alguno de los tamaños? En caso afirmativo, ¿a qué se debe este error? (Incorporar volcados de pantalla como se indica en las normas de prácticas)

RESPUESTA:

Se obtiene error para 6000000, este error se debe a que hemos agotado la pila usando variables locales

[illegible]

8. Generar los ejecutables del código fuente C para vectores globales y para dinámicos. Ejecutar los dos códigos en un nodo de cómputo de atcgrid usando un script como el del Listado 2 para el mismo rango de tamaños utilizado en el ejercicio anterior. Hay que poner en el script el nombre de los ficheros ejecutables generados en este ejercicio. Ejecutar también los códigos en el PC. ¿Se obtiene error usando vectores globales o dinámicos? ¿A qué cree que es debido? (Incorporar volcados de pantalla como se indica en las normas de prácticas)

RESPUESTA:

```

arturo@miior-pc: ~/bp0/ejer8
C2estudiante3@atcgird:~/bp0/ejer8 85x46
[ArturoCortesSanchez C2estudiante3@atcgird:~/bp0/ejer8] 2019-03-05 martes
$ qsub listado2.sh
7016.atcgird
[ArturoCortesSanchez C2estudiante3@atcgird:~/bp0/ejer8] 2019-03-05 martes
$ ls -lag
total 60
drwxrwxr-x 2 C2estudiante3 4096 mar 5 14:11 .
drwxrwxr-x 6 C2estudiante3 4096 feb 27 16:57 ..
-rw-r----- 1 C2estudiante3 0 mar 5 14:10 helloomp.e7016
-rw-r----- 1 C2estudiante3 2813 mar 5 14:10 helloomp.o7016
-rw-r--r-- 1 C2estudiante3 1133 mar 5 14:10 listado2.sh
-rwxrwxrwx 1 C2estudiante3 1220 mar 5 13:54 script_helloomp.sh
-rwxr-xr-x 1 C2estudiante3 16952 mar 5 13:50 sumaVectores_dynamic
-rwxr-xr-x 1 C2estudiante3 16928 mar 5 13:50 sumaVectores_global
[ArturoCortesSanchez C2estudiante3@atcgird:~/bp0/ejer8] 2019-03-05 martes
$ nano listado2.sh
[ArturoCortesSanchez C2estudiante3@atcgird:~/bp0/ejer8] 2019-03-05 martes
$ qsub listado2.sh
7025.atcgird
[ArturoCortesSanchez C2estudiante3@atcgird:~/bp0/ejer8] 2019-03-05 martes
$ ls -lag
total 64
drwxrwxr-x 2 C2estudiante3 4096 mar 5 14:20 .
drwxrwxr-x 6 C2estudiante3 4096 feb 27 16:57 ..
-rw-r----- 1 C2estudiante3 0 mar 5 14:10 helloomp.e7016
-rw-r----- 1 C2estudiante3 2813 mar 5 14:10 helloomp.o7016
-rw-r----- 1 C2estudiante3 2816 mar 5 14:20 helloomp.o7025
-rw-r--r-- 1 C2estudiante3 1134 mar 5 14:20 listado2.sh
-rwxrwxrwx 1 C2estudiante3 1220 mar 5 13:54 script_helloomp.sh
-rwxr-xr-x 1 C2estudiante3 16952 mar 5 13:50 sumaVectores_dynamic
-rwxr-xr-x 1 C2estudiante3 16928 mar 5 13:50 sumaVectores_global
[ArturoCortesSanchez C2estudiante3@atcgird:~/bp0/ejer8] 2019-03-05 martes
$

arturo@miior-pc: ~/bp0/ejer8
arturo@miior-pc: ~/bp0/ejer8 105x5
Fetching /home/C2estudiante3/bp0/ejer8/helloomp.o7016 to helloomp.o7016
/home/C2estudiante3/bp0/ejer8/helloomp.o7016 100% 2813 239.0KB/s 00:00
Fetching /home/C2estudiante3/bp0/ejer8/helloomp.o7025 to helloomp.o7025
/home/C2estudiante3/bp0/ejer8/helloomp.o7025 100% 2816 185.5KB/s 00:00
sftp>

arturo@miior-pc:~/bp0/ejer8 105x39
Nodo que ejecuta qsub: atcgird
Nodos asignados al trabajo:
atcgird3
Directorio de trabajo: /home/C2estudiante3/bp0/ejer8
Tamaño Vectores:65536 (4 B)
Tiempo:0.000473195 / Tamaño Vectores:65536 / V1[0]+V2[0]-V3[0](6553.600000+6553.600000=13107.200000) / / V1[65535]+V2[65535]-V3[65535](13107.100000+0.100000=13107.200000) /
Tamaño Vectores:131072 (4 B)
Tiempo:0.000937305 / Tamaño Vectores:131072 / V1[0]+V2[0]-V3[0](13107.200000+13107.200000=26214.400000) / / V1[131071]+V2[131071]-V3[131071](26214.300000+0.100000=26214.400000) /
Tamaño Vectores:262144 (4 B)
Tiempo:0.001895795 / Tamaño Vectores:262144 / V1[0]+V2[0]-V3[0](26214.400000+26214.400000=52428.800000) / / V1[262143]+V2[262143]-V3[262143](52428.700000+0.100000=52428.800000) /
Tamaño Vectores:524288 (4 B)
Tiempo:0.002856446 / Tamaño Vectores:524288 / V1[0]+V2[0]-V3[0](52428.800000+52428.800000=104857.600000) / / V1[524287]+V2[524287]-V3[524287](104857.500000+0.100000=104857.600000) /
Tamaño Vectores:1048576 (4 B)
Tiempo:0.005327766 / Tamaño Vectores:1048576 / V1[0]+V2[0]-V3[0](104857.600000+104857.600000=209715.200000) / / V1[1048575]+V2[1048575]-V3[1048575](209715.100000+0.100000=209715.200000) /
Tamaño Vectores:2097152 (4 B)
Tiempo:0.009479246 / Tamaño Vectores:2097152 / V1[0]+V2[0]-V3[0](209715.200000+209715.200000=419430.400000) / / V1[2097151]+V2[2097151]-V3[2097151](419430.300000+0.100000=419430.400000) /
Tamaño Vectores:4194304 (4 B)
Tiempo:0.017131838 / Tamaño Vectores:4194304 / V1[0]+V2[0]-V3[0](419430.400000+419430.400000=838860.800000) / / V1[4194303]+V2[4194303]-V3[4194303](838860.700000+0.100000=838860.800000) /
Tamaño Vectores:8388608 (4 B)
Tiempo:0.032999772 / Tamaño Vectores:8388608 / V1[0]+V2[0]-V3[0](838860.800000+838860.800000=1677721.600000) / / V1[8388607]+V2[8388607]-V3[8388607](1677721.500000+0.100000=1677721.600000) /
Tamaño Vectores:16777216 (4 B)
Tiempo:0.065804253 / Tamaño Vectores:16777216 / V1[0]+V2[0]-V3[0](1677721.600000+1677721.600000=3355443.200000) / / V1[16777215]+V2[16777215]-V3[16777215](3355443.100000+0.100000=3355443.200000) /
Tamaño Vectores:33554432 (4 B)
Tiempo:0.129977449 / Tamaño Vectores:33554432 / V1[0]+V2[0]-V3[0](3355443.200000+3355443.200000=6710886.400000) / / V1[33554431]+V2[33554431]-V3[33554431](6710886.300000+0.100000=6710886.400000) /
Tamaño Vectores:67108864 (4 B)
Tiempo:0.259687691 / Tamaño Vectores:67108864 / V1[0]+V2[0]-V3[0](6710886.400000+6710886.400000=13421772.800000) / / V1[67108863]+V2[67108863]-V3[67108863](13421772.700000+0.100000=13421772.800000) /
[ArturoCortesSanchez arturo@miior-pc:~/bp0/ejer8] 2019-03-05 martes
$

```

```

arturo@miior-pc: ~/bp0/ejer8
C2estudiante3@atcgird:~/bp0/ejer8 85x46
[ArturoCortesSanchez C2estudiante3@atcgird:~/bp0/ejer8] 2019-03-05 martes
$ qsub listado2.sh
7016.atcgird
[ArturoCortesSanchez C2estudiante3@atcgird:~/bp0/ejer8] 2019-03-05 martes
$ ls -lag
total 60
drwxrwxr-x 2 C2estudiante3 4096 mar 5 14:11 .
drwxrwxr-x 6 C2estudiante3 4096 feb 27 16:57 ..
-rw-r----- 1 C2estudiante3 0 mar 5 14:10 helloomp.e7016
-rw-r----- 1 C2estudiante3 2813 mar 5 14:10 helloomp.o7016
-rw-r--r-- 1 C2estudiante3 1133 mar 5 14:10 listado2.sh
-rwxrwxrwx 1 C2estudiante3 1220 mar 5 13:54 script_helloomp.sh
-rwxr-xr-x 1 C2estudiante3 16952 mar 5 13:50 sumaVectores_dynamic
-rwxr-xr-x 1 C2estudiante3 16928 mar 5 13:50 sumaVectores_global
[ArturoCortesSanchez C2estudiante3@atcgird:~/bp0/ejer8] 2019-03-05 martes
$

arturo@miior-pc: ~/bp0/ejer8
arturo@miior-pc: ~/bp0/ejer8 105x5
sftp> get helloomp.*
Fetching /home/C2estudiante3/bp0/ejer8/helloomp.e7016 to helloomp.e7016
Fetching /home/C2estudiante3/bp0/ejer8/helloomp.o7016 to helloomp.o7016
/home/C2estudiante3/bp0/ejer8/helloomp.o7016 100% 2813 308.7KB/s 00:00
sftp>

arturo@miior-pc:~/bp0/ejer8 105x39
$ cat helloomp.o7016
Id. usuario del trabajo: C2estudiante3
Id. del trabajo: 7016.atcgird
Nombre del trabajo especificado por usuario: helloomp
Directorio en el que se ha ejecutado qsub: /home/C2estudiante3/bp0/ejer8
Directorio de trabajo: /home/C2estudiante3
Cola: ac
Nodo que ejecuta qsub: atcgird
Nodos asignados al trabajo:
atcgird3
Directorio de trabajo: /home/C2estudiante3/bp0/ejer8
Tamaño Vectores:65536 (4 B)
Tiempo:0.000551472 / Tamaño Vectores:65536 / V1[0]+V2[0]-V3[0](6553.600000+6553.600000=13107.200000) / / V1[65535]+V2[65535]-V3[65535](13107.100000+0.100000=13107.200000) /
Tamaño Vectores:131072 (4 B)
Tiempo:0.000695198 / Tamaño Vectores:131072 / V1[0]+V2[0]-V3[0](13107.200000+13107.200000=26214.400000) / / V1[131071]+V2[131071]-V3[131071](26214.300000+0.100000=26214.400000) /
Tamaño Vectores:262144 (4 B)
Tiempo:0.001438781 / Tamaño Vectores:262144 / V1[0]+V2[0]-V3[0](26214.400000+26214.400000=52428.800000) / / V1[262143]+V2[262143]-V3[262143](52428.700000+0.100000=52428.800000) /
Tamaño Vectores:524288 (4 B)
Tiempo:0.002572667 / Tamaño Vectores:524288 / V1[0]+V2[0]-V3[0](52428.800000+52428.800000=104857.600000) / / V1[524287]+V2[524287]-V3[524287](104857.500000+0.100000=104857.600000) /
Tamaño Vectores:1048576 (4 B)
Tiempo:0.005573241 / Tamaño Vectores:1048576 / V1[0]+V2[0]-V3[0](104857.600000+104857.600000=209715.200000) / / V1[1048575]+V2[1048575]-V3[1048575](209715.100000+0.100000=209715.200000) /
Tamaño Vectores:2097152 (4 B)
Tiempo:0.009854475 / Tamaño Vectores:2097152 / V1[0]+V2[0]-V3[0](209715.200000+209715.200000=419430.400000) / / V1[2097151]+V2[2097151]-V3[2097151](419430.300000+0.100000=419430.400000) /
Tamaño Vectores:4194304 (4 B)
Tiempo:0.018069003 / Tamaño Vectores:4194304 / V1[0]+V2[0]-V3[0](419430.400000+419430.400000=838860.800000) / / V1[4194303]+V2[4194303]-V3[4194303](838860.700000+0.100000=838860.800000) /
Tamaño Vectores:8388608 (4 B)
Tiempo:0.034544769 / Tamaño Vectores:8388608 / V1[0]+V2[0]-V3[0](838860.800000+838860.800000=1677721.600000) / / V1[8388607]+V2[8388607]-V3[8388607](1677721.500000+0.100000=1677721.600000) /
Tamaño Vectores:16777216 (4 B)
Tiempo:0.067947684 / Tamaño Vectores:16777216 / V1[0]+V2[0]-V3[0](1677721.600000+1677721.600000=3355443.200000) / / V1[16777215]+V2[16777215]-V3[16777215](3355443.100000+0.100000=3355443.200000) /
Tamaño Vectores:33554432 (4 B)
Tiempo:0.135636113 / Tamaño Vectores:33554432 / V1[0]+V2[0]-V3[0](3355443.200000+3355443.200000=6710886.400000) / / V1[33554431]+V2[33554431]-V3[33554431](6710886.300000+0.100000=6710886.400000) /
Tamaño Vectores:67108864 (4 B)
Tiempo:0.271272229 / Tamaño Vectores:67108864 / V1[0]+V2[0]-V3[0](6710886.400000+6710886.400000=13421772.800000) / / V1[67108863]+V2[67108863]-V3[67108863](13421772.700000+0.100000=13421772.800000) /
[ArturoCortesSanchez arturo@miior-pc:~/bp0/ejer8] 2019-03-05 martes
$

```



```

arturo@miior-pc: ~/bp0/ejer8
arturo@miior-pc: ~/bp0/ejer8 192x32

[arturo@miior-pc: ejer8]$ cat ejecucionpc_dynamic.txt
Tamaño Vectores:65536 (4 B)
Tiempo:0.000760880 / Tamaño Vectores:65536 / V1[0]+V2[0]=V3[0] (6553.600000+6553.600000=13107.200000) / / V1[65535]+V2[65535]=V3[65535] (13107.100000+0.100000=13107.200000) /
Tamaño Vectores:131072 (4 B)
Tiempo:0.001631779 / Tamaño Vectores:131072 / V1[0]+V2[0]=V3[0] (13107.200000+13107.200000=26214.400000) / / V1[131071]+V2[131071]=V3[131071] (26214.300000+0.100000=26214.400000) /
Tamaño Vectores:262144 (4 B)
Tiempo:0.002867315 / Tamaño Vectores:262144 / V1[0]+V2[0]=V3[0] (26214.400000+26214.400000=52428.800000) / / V1[262143]+V2[262143]=V3[262143] (52428.700000+0.100000=52428.800000) /
Tamaño Vectores:524288 (4 B)
Tiempo:0.005190653 / Tamaño Vectores:524288 / V1[0]+V2[0]=V3[0] (52428.800000+52428.800000=104857.600000) / / V1[524287]+V2[524287]=V3[524287] (104857.500000+0.100000=104857.600000) /
Tamaño Vectores:1048576 (4 B)
Tiempo:0.008018768 / Tamaño Vectores:1048576 / V1[0]+V2[0]=V3[0] (104857.600000+104857.600000=209715.200000) / / V1[1048575]+V2[1048575]=V3[1048575] (209715.100000+0.100000=209715.200000) /
Tamaño Vectores:2097152 (4 B)
Tiempo:0.014785864 / Tamaño Vectores:2097152 / V1[0]+V2[0]=V3[0] (209715.200000+209715.200000=419430.400000) / / V1[2097151]+V2[2097151]=V3[2097151] (419430.300000+0.100000=419430.400000) /
Tamaño Vectores:4194304 (4 B)
Tiempo:0.029189701 / Tamaño Vectores:4194304 / V1[0]+V2[0]=V3[0] (419430.400000+419430.400000=838860.800000) / / V1[4194303]+V2[4194303]=V3[4194303] (838860.700000+0.100000=838860.800000) /
Tamaño Vectores:8388608 (4 B)
Tiempo:0.056620618 / Tamaño Vectores:8388608 / V1[0]+V2[0]=V3[0] (838860.800000+838860.800000=1677721.600000) / / V1[8388607]+V2[8388607]=V3[8388607] (1677721.500000+0.100000=1677721.600000) /
Tamaño Vectores:16777216 (4 B)
Tiempo:0.112104243 / Tamaño Vectores:16777216 / V1[0]+V2[0]=V3[0] (1677721.600000+1677721.600000=3355443.200000) / / V1[16777215]+V2[16777215]=V3[16777215] (3355443.100000+0.100000=3355443.200000) /
Tamaño Vectores:33554432 (4 B)
Tiempo:0.220165673 / Tamaño Vectores:33554432 / V1[0]+V2[0]=V3[0] (3355443.200000+3355443.200000=6710886.400000) / / V1[33554431]+V2[33554431]=V3[33554431] (6710886.300000+0.100000=6710886.400000) /
Tamaño Vectores:67108864 (4 B)
Tiempo:0.613059518 / Tamaño Vectores:67108864 / V1[0]+V2[0]=V3[0] (6710886.400000+6710886.400000=13421772.800000) / / V1[67108863]+V2[67108863]=V3[67108863] (13421772.700000+0.100000=13421772.800000) /
[arturo@miior-pc: ejer8]$

```

```

arturo@miior-pc: ~/bp0/ejer8
arturo@miior-pc: ~/bp0/ejer8 192x32

[arturo@miior-pc: ejer8]$ cat ejecucionpc_global.txt
Tamaño Vectores:65536 (4 B)
Tiempo:0.000599537 / Tamaño Vectores:65536 / V1[0]+V2[0]=V3[0] (6553.600000+6553.600000=13107.200000) / / V1[65535]+V2[65535]=V3[65535] (13107.100000+0.100000=13107.200000) /
Tamaño Vectores:131072 (4 B)
Tiempo:0.001603013 / Tamaño Vectores:131072 / V1[0]+V2[0]=V3[0] (13107.200000+13107.200000=26214.400000) / / V1[131071]+V2[131071]=V3[131071] (26214.300000+0.100000=26214.400000) /
Tamaño Vectores:262144 (4 B)
Tiempo:0.003331460 / Tamaño Vectores:262144 / V1[0]+V2[0]=V3[0] (26214.400000+26214.400000=52428.800000) / / V1[262143]+V2[262143]=V3[262143] (52428.700000+0.100000=52428.800000) /
Tamaño Vectores:524288 (4 B)
Tiempo:0.005533737 / Tamaño Vectores:524288 / V1[0]+V2[0]=V3[0] (52428.800000+52428.800000=104857.600000) / / V1[524287]+V2[524287]=V3[524287] (104857.500000+0.100000=104857.600000) /
Tamaño Vectores:1048576 (4 B)
Tiempo:0.011239902 / Tamaño Vectores:1048576 / V1[0]+V2[0]=V3[0] (104857.600000+104857.600000=209715.200000) / / V1[1048575]+V2[1048575]=V3[1048575] (209715.100000+0.100000=209715.200000) /
Tamaño Vectores:2097152 (4 B)
Tiempo:0.019817531 / Tamaño Vectores:2097152 / V1[0]+V2[0]=V3[0] (209715.200000+209715.200000=419430.400000) / / V1[2097151]+V2[2097151]=V3[2097151] (419430.300000+0.100000=419430.400000) /
Tamaño Vectores:4194304 (4 B)
Tiempo:0.029750759 / Tamaño Vectores:4194304 / V1[0]+V2[0]=V3[0] (419430.400000+419430.400000=838860.800000) / / V1[4194303]+V2[4194303]=V3[4194303] (838860.700000+0.100000=838860.800000) /
Tamaño Vectores:8388608 (4 B)
Tiempo:0.054745774 / Tamaño Vectores:8388608 / V1[0]+V2[0]=V3[0] (838860.800000+838860.800000=1677721.600000) / / V1[8388607]+V2[8388607]=V3[8388607] (1677721.500000+0.100000=1677721.600000) /
Tamaño Vectores:16777216 (4 B)
Tiempo:0.108299915 / Tamaño Vectores:16777216 / V1[0]+V2[0]=V3[0] (1677721.600000+1677721.600000=3355443.200000) / / V1[16777215]+V2[16777215]=V3[16777215] (3355443.100000+0.100000=3355443.200000) /
Tamaño Vectores:33554432 (4 B)
Tiempo:0.221667232 / Tamaño Vectores:33554432 / V1[0]+V2[0]=V3[0] (3355443.200000+3355443.200000=6710886.400000) / / V1[33554431]+V2[33554431]=V3[33554431] (6710886.300000+0.100000=6710886.400000) /
Tamaño Vectores:67108864 (4 B)
Tiempo:0.214778597 / Tamaño Vectores:33554432 / V1[0]+V2[0]=V3[0] (3355443.200000+3355443.200000=6710886.400000) / / V1[33554431]+V2[33554431]=V3[33554431] (6710886.300000+0.100000=6710886.400000) /
[arturo@miior-pc: ejer8]$

```

No se obtiene ningún error en la ejecución dinámica, pero en la ejecución global el tamaño del vector hace tope en 33554431

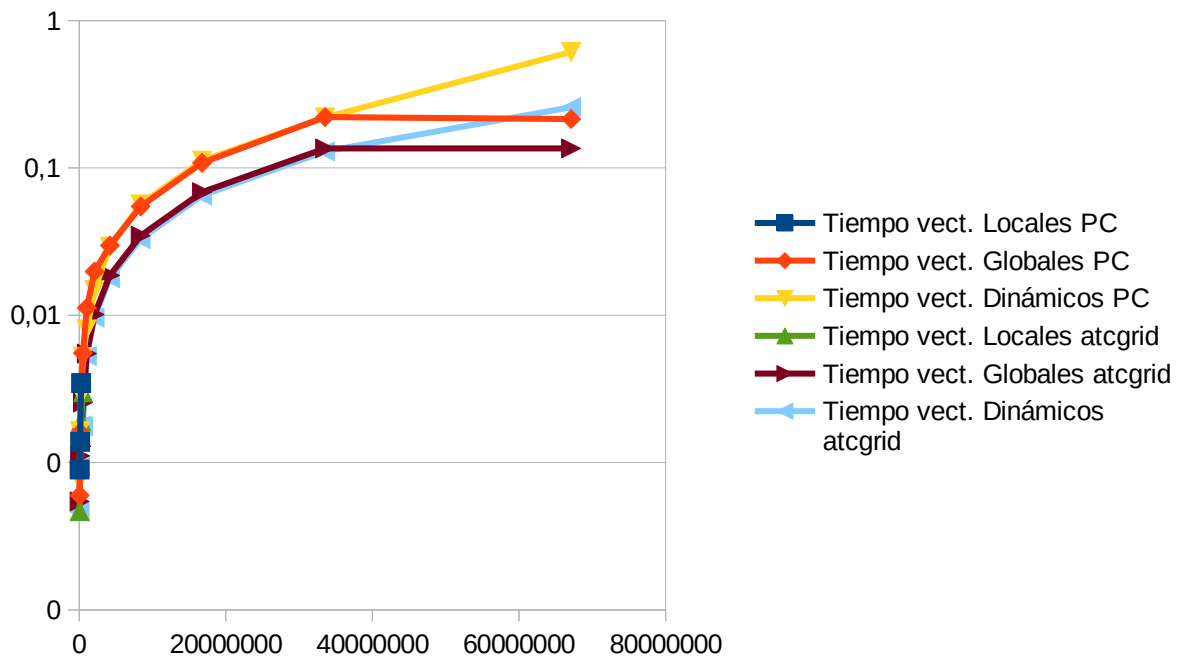
9. Rellenar una tabla como la Tabla 1 **en una hoja de cálculo** con los tiempos de ejecución obtenidos en los ejercicios anteriores para el trozo de código que realiza la suma de vectores. Debe haber una tabla para atcgrid y otra para su PC en la hoja de cálculo. En la columna “Bytes de un vector” hay que poner el total de bytes reservado para un vector. Con ayuda de la hoja de cálculo representar **en una misma gráfica** los tiempos de ejecución obtenidos en atcgrid y en su PC para vectores locales, globales y dinámicos (eje y) en función del tamaño en bytes de un vector (por tanto, los valores de la segunda columna de la tabla, que están en escala logarítmica, deben estar en el eje x). Utilizar escala logarítmica en el eje de ordenadas (eje y). (NOTA: Se recomienda usar en la hoja de cálculo el mismo separador para decimales que usan los códigos. Este separador se puede modificar en la hoja de cálculo.)

(a) Copiar las tablas y la gráfica en el cuaderno de prácticas.

RESPUESTA:

Nº de Componentes	Bytes de un vector	Tiempo vect. Locales PC	Tiempo vect. Globales PC	Tiempo vect. Dinámicos PC
65536	262144	0,000896469	0,000599537	0,000760880
131072	524288	0,001380714	0,001603013	0,001631779
262144	1048576	0,003463867	0,003331460	0,002867315
524288	2097152		0,005533737	0,005190653
1048576	4194304		0,011239902	0,008018768
2097152	8388608		0,019817531	0,014785864
4194304	16777216		0,029750759	0,029189701
8388608	33554432		0,054745774	0,056620618
16777216	67108864		0,108299915	0,112104243
33554432	134217728		0,221667232	0,220165673
67108864	268435456		0,214778597	0,613059518

Nº de Componentes	Bytes de un vector	Tiempo vect. Locales atcgrid	Tiempo vect. Globales atcgrid	Tiempo vect. Dinámicos atcgrid
65536	262144	0,000468201	0,000543773	0,000462123
131072	524288	0,000948806	0,001109740	0,000936020
262144	1048576	0,001906180	0,001290322	0,001784948
524288	2097152	0,003003633	0,002558185	0,001763581
1048576	4194304		0,005491028	0,005296979
2097152	8388608		0,010123719	0,009638142
4194304	16777216		0,018639315	0,017838803
8388608	33554432		0,034706493	0,032959354
16777216	67108864		0,068423513	0,064993738
33554432	134217728		0,135752449	0,129650488
67108864	268435456		0,135689100	0,258662403



10.

(b) ¿Hay diferencias en los tiempos de ejecución?

RESPUESTA:

No hay a penas diferencias de tiempos de ejecución entre los distintos vectores, pero el tiempo de ejecución en atcgrid es claramente inferior al de mi PC

11. (a) ¿Cuál es el máximo valor que se puede almacenar en la variable N teniendo en cuenta su tipo? Razonar respuesta.

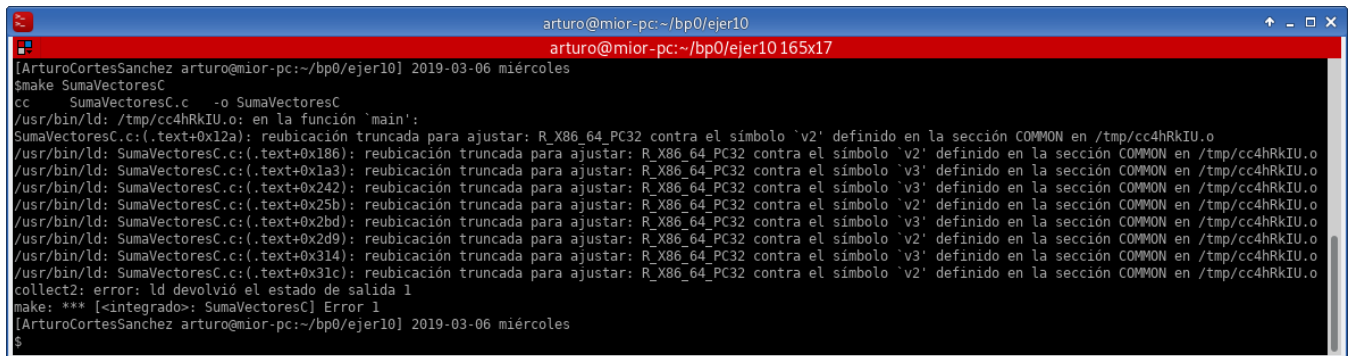
RESPUESTA:

Un dato tipo int está formado por 32 bits, por tanto el valor máximo de la variable es $2^{32}-1=4294967295$

- (b) Modificar el código fuente C (en el PC) 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 y generar el ejecutable. ¿Qué ocurre? ¿A qué es debido? (Incorporar volcados de pantalla que muestren lo que ocurre)

RESPUESTA:

Obtenemos el siguiente error de enlazado:



```

arturo@mior-pc:~/bp0/ejer10
arturo@mior-pc:~/bp0/ejer10 165x17
[ArturoCortesSanchez arturo@mior-pc:~/bp0/ejer10] 2019-03-06 miércoles
$make SumaVectoresC
cc SumaVectoresC.c -o SumaVectoresC
/usr/bin/ld: /tmp/cc4hRkIU.o: en la función `main':
SumaVectoresC.c:(.text+0x12a): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo `v2' definido en la sección COMMON en /tmp/cc4hRkIU.o
/usr/bin/ld: SumaVectoresC.c:(.text+0x186): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo `v2' definido en la sección COMMON en /tmp/cc4hRkIU.o
/usr/bin/ld: SumaVectoresC.c:(.text+0x1a3): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo `v3' definido en la sección COMMON en /tmp/cc4hRkIU.o
/usr/bin/ld: SumaVectoresC.c:(.text+0x242): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo `v3' definido en la sección COMMON en /tmp/cc4hRkIU.o
/usr/bin/ld: SumaVectoresC.c:(.text+0x25b): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo `v2' definido en la sección COMMON en /tmp/cc4hRkIU.o
/usr/bin/ld: SumaVectoresC.c:(.text+0x2bd): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo `v3' definido en la sección COMMON en /tmp/cc4hRkIU.o
/usr/bin/ld: SumaVectoresC.c:(.text+0x2d9): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo `v2' definido en la sección COMMON en /tmp/cc4hRkIU.o
/usr/bin/ld: SumaVectoresC.c:(.text+0x314): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo `v3' definido en la sección COMMON en /tmp/cc4hRkIU.o
/usr/bin/ld: SumaVectoresC.c:(.text+0x31c): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo `v2' definido en la sección COMMON en /tmp/cc4hRkIU.o
collect2: error: ld devolvió el estado de salida 1
make: *** [: SumaVectoresC] Error 1
[ArturoCortesSanchez arturo@mior-pc:~/bp0/ejer10] 2019-03-06 miércoles
$

```

Esto es debido a que al hacer este cambio hemos excedido el tamaño máximo del data segment.

Entrega del trabajo

Leer lo indicado en las normas de prácticas sobre la entrega del trabajo del bloque práctico en SWAD.

Listado 1. Código C que suma dos vectores

```

/* SumaVectoresC.c
   Suma de dos vectores: v3 = v1 + v2

   Para compilar usar (-lrt: real time library, no todas las versiones de gcc necesitan que se incluya
   -lrt):
       gcc -O2 SumaVectores.c -o SumaVectores -lrt
       gcc -O2 -S SumaVectores.c -lrt    //para generar el código ensamblador

   Para ejecutar use: SumaVectoresC longitud
*/

#include <stdlib.h> // biblioteca con funciones atoi(), malloc() y free()
#include <stdio.h>  // biblioteca donde se encuentra la función printf()
#include <time.h>    // biblioteca donde se encuentra la función clock_gettime()

//Sólo puede estar definida una de las tres constantes VECTOR_ (sólo uno de los ...
//tres defines siguientes puede estar descomentado):
//#define VECTOR_LOCAL    // descomentar para que los vectores sean variables ...
//                          // locales (si se supera el tamaño de la pila se ...
//                          // generará el error "Violación de Segmento")
//#define VECTOR_GLOBAL// descomentar para que los vectores sean variables ...
//                          // globales (su longitud no estará limitada por el ...
//                          // tamaño de la pila del programa)
#define VECTOR_DYNAMIC    // descomentar para que los vectores sean variables ...
//                          // dinámicas (memoria reutilizable durante la ejecución)

#ifdef VECTOR_GLOBAL
#define MAX 33554432        //2^25
double v1[MAX], v2[MAX], v3[MAX];
#endif

int main(int argc, char** argv){

    int i;
    struct timespec cgt1,cgt2; double ncgt; //para tiempo de ejecución

    //Leer argumento de entrada (nº de componentes del vector)
    if (argc<2){
        printf("Faltan nº componentes del vector\n");
        exit(-1);
    }

    unsigned int N = atoi(argv[1]); // Máximo N =2^32-1=4294967295 (sizeof(unsigned int) = 4 B)
    #ifdef VECTOR_LOCAL
        double v1[N], v2[N], v3[N];    // Tamaño variable local en tiempo de ejecución ...
                                        // disponible en C a partir de actualización C99
    #endif
    #ifdef VECTOR_GLOBAL
        if (N>MAX) N=MAX;
    #endif
    #ifdef VECTOR_DYNAMIC
        double *v1, *v2, *v3;
        v1 = (double*) malloc(N*sizeof(double)); // malloc necesita el tamaño en bytes
        v2 = (double*) malloc(N*sizeof(double)); //si no hay espacio suficiente malloc devuelve NULL
        v3 = (double*) malloc(N*sizeof(double));

```

```

    if ( (v1==NULL) || (v2==NULL) || (v3==NULL) ){
        printf("Error en la reserva de espacio para los vectores\n");
        exit(-2);
    }
#endif

//Inicializar vectores
for(i=0; i<N; i++){
    v1[i] = N*0.1+i*0.1; v2[i] = N*0.1-i*0.1; //los valores dependen de N
}

clock_gettime(CLOCK_REALTIME,&cgt1);
//Calcular suma de vectores
for(i=0; i<N; i++)
    v3[i] = v1[i] + v2[i];

clock_gettime(CLOCK_REALTIME,&cgt2);
ncgt=(double) (cgt2.tv_sec-cgt1.tv_sec)+
        (double) ((cgt2.tv_nsec-cgt1.tv_nsec)/(1.e+9));

//Imprimir resultado de la suma y el tiempo de ejecución
if (N<10) {
    printf("Tiempo(seg.):%11.9f\t / Tamaño Vectores:%u\n",ncgt,N);
    for(i=0; i<N; i++)
        printf("/ V1[%d]+V2[%d]=V3[%d](%8.6f+%8.6f=%8.6f) /\n",
            i,i,i,v1[i],v2[i],v3[i]);
}
else
    printf("Tiempo(seg.):%11.9f\t / Tamaño Vectores:%u\t/ V1[0]+V2[0]=V3[0](%8.6f+%8.6f=%8.6f) / /
        V1[%d]+V2[%d]=V3[%d](%8.6f+%8.6f=%8.6f) /\n",
        ncgt,N,v1[0],v2[0],v3[0],N-1,N-1,N-1,v1[N-1],v2[N-1],v3[N-1]);

#ifdef VECTOR_DYNAMIC
free(v1); // libera el espacio reservado para v1
free(v2); // libera el espacio reservado para v2
free(v3); // libera el espacio reservado para v3
#endif
return 0;
}

```

Listado 2 . Script para la suma de vectores (SumaVectores.sh). Se supone en el script que el fichero a ejecutar se llama SumaVectorC.

```

#!/bin/bash
#Todos los scripts que se hagan para atcgrid deben incluir lo siguiente:
#Se asigna al trabajo el nombre SumaVectoresC_vlocales
#PBS -N SumaVectoresC_vlocales
#Se asigna al trabajo la cola ac
#PBS -q ac
#Se imprime información del trabajo usando variables de entorno de PBS
echo "Id. usuario del trabajo: $PBS_O_LOGNAME"
echo "Id. del trabajo: $PBS_JOBID"

```

```
echo "Nombre del trabajo especificado por usuario: $PBS_JOBNAME"
echo "Nodo que ejecuta qsub: $PBS_O_HOST"
echo "Directorio en el que se ha ejecutado qsub: $PBS_O_WORKDIR"
echo "Cola: $PBS_QUEUE"
echo "Nodos asignados al trabajo:"
cat $PBS_NODEFILE
# FIN del trozo que deben incluir todos los scripts

#para N potencia de 2 desde 2^16 a 2^26
for ((N=65536;N<67108865;N=N*2))
do
    Poner_el_camino_al_ejecutable/SumaVectoresC $N
done
```