

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): Francisco Javier Bolívar Lupiáñez

Grupo de prácticas: B1

Fecha de entrega: 11/03/2014 (hasta las 12 pm)

Fecha evaluación en clase: 19/03/2014

1. En el primer ejemplo de ejecución en atcgrid usando TORQUE se ejecuta el ejemplo HelloOMP.c de la página 12 del seminario usando la siguiente orden: `echo 'hello/HelloOMP' | qsub -q ac`. El resultado de la ejecución de este código en atcgrid se puede ver en la página 19 del seminario. Conteste a las siguientes preguntas:

- a. ¿Para qué se usa en `qsub` la opción `-q`?

**RESPUESTA:** Para mandarlo a una cola específica.

- b. ¿Cómo sabe el usuario que ha terminado la ejecución en atcgrid?

**RESPUESTA:** Si utilizamos `qstat` podemos ver en una columna el estado del envío (s), si aparece C, significa que se ha completado.

- c. ¿Cómo puede saber el usuario si ha habido algún error en la ejecución?

**RESPUESTA:** Si ejecutas `ls -lag` puedes ver el tamaño de los ficheros. Si el `.e` tiene valor 0, no habrá habido fallos.

- d. ¿Cómo ve el usuario el resultado de la ejecución?

**RESPUESTA:** Haciendo `cat` sobre el fichero `.o` o abriéndolo desde un gestor de archivos.

- e. ¿Por qué en el resultado de la ejecución aparecen 24 saludos “`!!!Hello World!!!`”?

**RESPUESTA:** Porque en atcgrid contamos con un total de 12 cores físicos, por tanto corresponde a 24 cores virtuales y al usar `pragma omp parallel`, se crean tantas hebras como cores virtuales contemos.

2. En el segundo ejemplo de ejecución en atcgrid usando TORQUE el script `script_helloomp.sh` de la página 22 del seminario usando la siguiente orden: `qsub script_helloomp.sh`. El script ejecuta varias veces el ejecutable del código HelloOMP.c. El resultado de la ejecución de este código en atcgrid se puede ver en la página 26 del seminario. Conteste a las siguientes preguntas:

- a. ¿Por qué no acompaña a al orden `qsub` la opción `-q` en este caso?

**RESPUESTA:** Porque estamos lanzando un guión que ejecuta esa operación y por tanto la incorpora.

- b. ¿Cuántas veces ejecuta el script el ejecutable HelloOMP en atcgrid?

**RESPUESTA:** 4, la primera vez para 12 hebras, la segunda para 6, la tercera para 3 y la última para 1.

- c. ¿Cuántos saludos “`!!!Hello World!!!`” se imprimen en cada ejecución? ¿Por qué se imprime ese número?

**RESPUESTA:** La primera vez 12, 6 la siguiente, 3 la siguiente y 1 la última. La primera vez imprime 12 porque hemos dicho que lance 12 hebras, después hemos continuado con el bucle haciendo en cada iteración la mitad, por eso en cada “vuelta” se reduce a la mitad el número de ejecuciones.

**3.** Realizar las siguientes modificaciones en el script “`!!!Hello World!!!`”:

- Eliminar la variable de entorno `$PBS_O_WORKDIR` en el punto en el que aparece.
- Añadir lo necesario para que, cuando se ejecute el script, se imprima la variable de entorno `$PBS_O_WORKDIR`.

Ejecutar el script con estas modificaciones. ¿Qué resultados de ejecución se obtienen en este caso? Incorporar en el cuaderno de trabajo volcados de pantalla que muestren estos resultados.

**RESPUESTA:**

```
[B1estudiante3@atcgrid hello]$ ls -laq
total 32
drwxrwxr-x 2 B1estudiante3 B1estudiante3 4096 feb 25 19:51 .
drwx----- 4 B1estudiante3 B1estudiante3 4096 feb 25 19:47 ..
-rwxrwxr-x 1 B1estudiante3 B1estudiante3 8609 feb 25 12:06 HelloMP
-rw----- 1 B1estudiante3 B1estudiante3 380 feb 25 2014 hellomp.e23056
-rw----- 1 B1estudiante3 B1estudiante3 491 feb 25 2014 hellomp.o23056
-rw-rw-r-- 1 B1estudiante3 B1estudiante3 913 feb 25 19:47 script_hellomp.sh
```

Podemos observar que ha habido errores en la ejecución porque el archivo `.e` no está vacío.

```
fblupi@fblupi-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC/Practicas/P0$ cat hellomp.e23056
/var/spool/torque/mom_priv/jobs/23056.atcgrid.SC: línea 24: HelloMP: no se encontró la orden
/var/spool/torque/mom_priv/jobs/23056.atcgrid.SC: línea 24: HelloMP: no se encontró la orden
/var/spool/torque/mom_priv/jobs/23056.atcgrid.SC: línea 24: HelloMP: no se encontró la orden
/var/spool/torque/mom_priv/jobs/23056.atcgrid.SC: línea 24: HelloMP: no se encontró la orden
```

El error está en que no encuentra el archivo, ya que al quitar `$PBS_O_WORKDIR` quitamos la ruta donde está alojada el archivo y no se puede encontrar. De esta forma vemos lo necesario que es incluirla, porque si no, a no ser que el script esté en el directorio raíz del usuario, no se llegará a ejecutar bien.

```
fblupi@fblupi-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC/Practicas/P0$ cat hellomp.o23056
Id. usuario del trabajo: B1estudiante3
Id. del trabajo: 23056.atcgrid
Nombre del trabajo especificado por usuario: hellomp
Nodo que ejecuta qsub: atcgrid
Cola: ac
Nodos asignados al trabajo:
atcgrid1
No de threads inicial: 12

Para 12 threads:
Directorio de trabajo: /home/B1estudiante3/hello

Para 6 threads:
Directorio de trabajo: /home/B1estudiante3/hello

Para 3 threads:
Directorio de trabajo: /home/B1estudiante3/hello

Para 1 threads:
Directorio de trabajo: /home/B1estudiante3/hello
```

En esta última imagen, con la salida del programa, podemos ver que efectivamente, esta variable contenía la ruta a la que nos referíamos.

**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`), del PC del aula de prácticas y de su PC (si tiene Linux instalado). Indique qué ha hecho para obtener el contenido de `/proc/cpuinfo` en atcgrid.

**RESPUESTA:**

processor 23 atcgrid

```
processor      : 23
vendor_id     : GenuineIntel
cpu family    : 6
model         : 44
model name    : Intel(R) Xeon(R) CPU           E5645  @ 2.40GHz
stepping      : 2
cpu MHz       : 1600.000
cache size    : 12288 KB
physical id   : 1
siblings      : 12
core id       : 10
cpu cores     : 6
apicid        : 53
initial apicid : 53
fpu           : yes
fpu_exception : yes
cpuid level   : 11
wp            : yes
flags         : 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 syscall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc aperfmperf pni pclmulqdq dtes64 monitor ds_cpl vmx est tm2 ssse3 cx16 xtpr pdcm dca sse4_1 sse4_2 popcnt lah_f_lm ida arat epb dts tpr_shadow vnmi flexpriority ept vpid
bogomips      : 4799.89
clflush size  : 64
cache alignment : 64
address sizes  : 40 bits physical, 48 bits virtual
power management:
```

processor 3 mi pc

```
processor      : 3
vendor_id     : GenuineIntel
cpu family    : 6
model         : 42
model name    : Intel(R) Core(TM) i3-2367M CPU @ 1.40GHz
stepping      : 7
microcode     : 0x28
cpu MHz       : 800.000
cache size    : 3072 KB
physical id   : 0
siblings      : 4
core id       : 1
cpu cores     : 2
apicid        : 3
initial apicid : 3
fpu           : yes
fpu_exception : yes
cpuid level   : 13
wp            : yes
flags         : 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 syscall nx rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc aperfmperf pni pclmulqdq dtes64 monitor ds_cpl vmx est tm2 ssse3 cx16 xtpr pdcm pcid sse4_1 sse4_2 x2apic popcnt tsc_deadline_timer xsave avx lahf_lm arat epb xsaveopt pln pts dtherm tpr_shadow vnmi flexpriority ept vpid
bogomips      : 2793.65
clflush size  : 64
cache alignment : 64
address sizes  : 36 bits physical, 48 bits virtual
power management:
```

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++:

$$v3 = v1 + v2; \quad v3(i) = v1(i) + v2(i), \quad i=0, \dots, N-1$$

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`.

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 devuelve la función `clock_gettime()`?

**RESPUESTA:** En `ncgt` calculamos el tiempo de ejecución con una precisión de nanosegundos. Con `clock_gettime()` se devuelve el tiempo actual, así que restando el tiempo final menos el inicial obtendremos el total transcurrido.



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:** La forma de reservar memoria para el vector dinámico. En c++ usa `new` y si no hay espacio suficiente genera una excepción. En c se utiliza `malloc` y si no hay espacio suficiente devuelve `NULL` por lo que inmediatamente después comprobamos si algún vector tiene valor `NULL` para parar el programa. Además de esto nos encontraríamos con diferencias propias de cada lenguaje.

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:**

```
[B1estudiante3@atcgrid hello]$ ls -l
total 56
-rw-rw-r-- 1 B1estudiante3 B1estudiante3 20358 mar  4 11:27 cpuinfo_atcgrid.txt
-rwxrwxr-x 1 B1estudiante3 B1estudiante3  8609 feb 25 20:04 HelloOMP
-rw----- 1 B1estudiante3 B1estudiante3    0 feb 25 20:36 helloomp.e23061
-rw----- 1 B1estudiante3 B1estudiante3   760 feb 25 20:36 helloomp.o23061
-rw-rw-r-- 1 B1estudiante3 B1estudiante3   882 feb 25 20:11 script_helloomp.sh
-rw----- 1 B1estudiante3 B1estudiante3    0 mar  4 2014 STDIN.e24985
-rw----- 1 B1estudiante3 B1estudiante3   151 mar  4 2014 STDIN.o24985
-rwxrwxr-x 1 B1estudiante3 B1estudiante3  8664 mar  4 12:23 SumaVectoresC
```

```
[B1estudiante3@atcgrid hello]$ cat STDIN.o24985
Tiempo(seg.):0.000000172      / Tamaño Vectores:14      / V1[0]+V2[0]=V3[0](1.4
00000+1.40000=2.80000) / / V1[13]+V2[13]=V3[13](2.700000+0.100000=2.800000) /
```

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:** Aparece error tras 262144, porque al multiplicarse otra vez por 2 se superaría el tamaño de la pila.

```
fblupi@fblupi-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC/Prac
ticas/P0$ cat SumaVectoresC_vlocales.o25037
Id. usuario del trabajo: B1estudiante3
Id. del trabajo: 25037.atcgrid
Nombre del trabajo especificado por usuario: SumaVectoresC_vlocales
Nodo que ejecuta qsub: atcgrid
Directorio en el que se ha ejecutado qsub: /home/B1estudiante3/hello
Cola: ac
Nodos asignados al trabajo:
atcgrid1
Tiempo(seg.):0.000438491      / 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) /
/
Tiempo(seg.):0.000882654      / 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=26214.400
000) /
Tiempo(seg.):0.001784521      / 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=52428.800
000) /
```

```
fblupi@fblupi-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC/Prac
ticas/P0$ cat SumaVectoresC_vlocales.e25037
/var/spool/torque/mom_priv/jobs/25037.atcgrid.SC: línea 20: 4873 Violación de segmento ('core'
generado) $PBS_O_WORKDIR/SumaVectoresC $N
/var/spool/torque/mom_priv/jobs/25037.atcgrid.SC: línea 20: 4876 Violación de segmento ('core'
generado) $PBS_O_WORKDIR/SumaVectoresC $N
/var/spool/torque/mom_priv/jobs/25037.atcgrid.SC: línea 20: 4879 Violación de segmento ('core'
generado) $PBS_O_WORKDIR/SumaVectoresC $N
/var/spool/torque/mom_priv/jobs/25037.atcgrid.SC: línea 20: 4881 Violación de segmento ('core'
generado) $PBS_O_WORKDIR/SumaVectoresC $N
/var/spool/torque/mom_priv/jobs/25037.atcgrid.SC: línea 20: 4883 Violación de segmento ('core'
generado) $PBS_O_WORKDIR/SumaVectoresC $N
/var/spool/torque/mom_priv/jobs/25037.atcgrid.SC: línea 20: 4885 Violación de segmento ('core'
generado) $PBS_O_WORKDIR/SumaVectoresC $N
/var/spool/torque/mom_priv/jobs/25037.atcgrid.SC: línea 20: 4887 Violación de segmento ('core'
generado) $PBS_O_WORKDIR/SumaVectoresC $N
/var/spool/torque/mom_priv/jobs/25037.atcgrid.SC: línea 20: 4889 Violación de segmento ('core'
generado) $PBS_O_WORKDIR/SumaVectoresC $N
```

No podría ejecutarlo en mi ordenador porque el .sh está hecho para atcgrid pero asignando los mismos tamaños manualmente aparecería el mismo error:

```
^[[5~fblupi@fblupi-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC
ticas/P0$ ./SumaVectoresC 65536
Tiempo(seg.):0.000651327 / 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)
/
fblupi@fblupi-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC/Prac
ticas/P0$ ./SumaVectoresC 131072
Tiempo(seg.):0.001354225 / 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=26214.400
000) /
fblupi@fblupi-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC/Prac
ticas/P0$ ./SumaVectoresC 262144
Tiempo(seg.):0.001770943 / 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=52428.800
000) /
fblupi@fblupi-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC/Prac
ticas/P0$ ./SumaVectoresC 524288
Violación de segmento
```

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:**

Ejecutamos ambos scripts y vemos que no hay errores.

```
[B1estudiante3@atcgrid hello]$ qsub SumaVectoresGlobal.sh
25312.atcgrid
```

```
[B1estudiante3@atcgrid hello]$ qsub SumaVectoresDynamic.sh
25315.atcgrid
```



```
[Blestudiante3@atcgrid hello]$ ls -l
total 124
-rw-rw-r-- 1 Blestudiante3 Blestudiante3 20358 mar  4 11:27 cpuinfo_atcgrid.txt
-rwxrwxr-x 1 Blestudiante3 Blestudiante3 8609 feb 25 20:04 HelloOMP
-rw----- 1 Blestudiante3 Blestudiante3 0 feb 25 20:36 helloomp.e23061
-rw----- 1 Blestudiante3 Blestudiante3 760 feb 25 20:36 helloomp.o23061
-rw-rw-r-- 1 Blestudiante3 Blestudiante3 882 feb 25 20:11 script_helloomp.sh
-rw----- 1 Blestudiante3 Blestudiante3 0 mar  4 12:51 STDIN.e24985
-rw----- 1 Blestudiante3 Blestudiante3 151 mar  4 12:51 STDIN.o24985
-rwxrwxr-x 1 Blestudiante3 Blestudiante3 8664 mar  4 12:23 SumaVectoresC
-rw----- 1 Blestudiante3 Blestudiante3 0 mar  4 2014 SumaVectoresC_vdinamicos.e25315
-rw----- 1 Blestudiante3 Blestudiante3 2436 mar  4 2014 SumaVectoresC_vdinamicos.o25315
-rw----- 1 Blestudiante3 Blestudiante3 0 mar  4 2014 SumaVectoresC_vglobales.e25312
-rw----- 1 Blestudiante3 Blestudiante3 2432 mar  4 2014 SumaVectoresC_vglobales.o25312
-rw----- 1 Blestudiante3 Blestudiante3 1128 mar  4 13:06 SumaVectoresC_vlocales.e25037
-rw----- 1 Blestudiante3 Blestudiante3 839 mar  4 13:06 SumaVectoresC_vlocales.o25037
-rwxrwxr-x 1 Blestudiante3 Blestudiante3 8706 mar  4 19:41 SumaVectoresDynamic
-rw-rw-r-- 1 Blestudiante3 Blestudiante3 753 mar  4 19:41 SumaVectoresDynamic.sh
-rwxrwxr-x 1 Blestudiante3 Blestudiante3 8685 mar  4 19:42 SumaVectoresGlobal
-rw-rw-r-- 1 Blestudiante3 Blestudiante3 750 mar  4 19:42 SumaVectoresGlobal.sh
-rwxrwxr-x 1 Blestudiante3 Blestudiante3 8664 mar  4 19:41 SumaVectoresLocal
-rw-rw-r-- 1 Blestudiante3 Blestudiante3 747 mar  4 19:41 SumaVectoresLocal.sh
-rw-rw-r-- 1 Blestudiante3 Blestudiante3 743 mar  4 12:41 SumaVectores.sh
```

```
[Blestudiante3@atcgrid hello]$ cat SumaVectoresC_vglobales.o25312
Id. usuario del trabajo: Blestudiante3
Id. del trabajo: 25312.atcgrid
Nombre del trabajo especificado por usuario: SumaVectoresC_vglobales
Modo que ejecuta qsub: atcgrid
Directorio en el que se ha ejecutado qsub: /home/Blestudiante3/hello
Cola: ac
Nodos asignados al trabajo:
atcgrid1
Tiempo(seg.):0.000549985 / 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) /
Tiempo(seg.):0.000534678 / Tamaño Vectores:131072 / V1[0]+V2[0]=V3[0](13107.200000+13107.200000=26214.400000) / V1[131071]+V2[131071]=V3[131071](26214.30
0000+0.100000=26214.400000) /
Tiempo(seg.):0.001436022 / Tamaño Vectores:262144 / V1[0]+V2[0]=V3[0](26214.400000+26214.400000=52428.800000) / V1[262143]+V2[262143]=V3[262143](52428.70
0000+0.100000=52428.800000) /
Tiempo(seg.):0.002640176 / 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) /
Tiempo(seg.):0.004801694 / Tamaño Vectores:1048576 / V1[0]+V2[0]=V3[0](104857.600000+104857.600000=209715.200000) / V1[1048575]+V2[1048575]=V3[1048575](20
9715.100000+0.100000=209715.200000) /
Tiempo(seg.):0.008574328 / Tamaño Vectores:2097152 / V1[0]+V2[0]=V3[0](209715.200000+209715.200000=419430.400000) / V1[2097151]+V2[2097151]=V3[2097151](41
9430.300000+0.100000=419430.400000) /
Tiempo(seg.):0.016399286 / Tamaño Vectores:4194304 / V1[0]+V2[0]=V3[0](419430.400000+419430.400000=838860.800000) / V1[4194303]+V2[4194303]=V3[4194303](83
8860.700000+0.100000=838860.800000) /
Tiempo(seg.):0.032141182 / Tamaño Vectores:8388608 / V1[0]+V2[0]=V3[0](838860.800000+838860.800000=1677721.600000) / V1[8388607]+V2[8388607]=V3[8388607](1
677721.500000+0.100000=1677721.600000) /
Tiempo(seg.):0.063612018 / Tamaño Vectores:16777216 / V1[0]+V2[0]=V3[0](1677721.600000+1677721.600000=3355443.200000) / V1[16777215]+V2[16777215]=V3[167772
15](3355443.100000+0.100000=3355443.200000) /
Tiempo(seg.):0.124973505 / Tamaño Vectores:33554432 / V1[0]+V2[0]=V3[0](3355443.200000+3355443.200000=6710886.400000) / V1[33554431]+V2[33554431]=V3[335544
31](6710886.300000+0.100000=6710886.400000) /
Tiempo(seg.):0.124966156 / Tamaño Vectores:33554432 / V1[0]+V2[0]=V3[0](3355443.200000+3355443.200000=6710886.400000) / V1[33554431]+V2[33554431]=V3[335544
31](6710886.300000+0.100000=6710886.400000) /
```

```
[Blestudiante3@atcgrid hello]$ cat SumaVectoresC_vdinamicos.o25315
Id. usuario del trabajo: Blestudiante3
Id. del trabajo: 25315.atcgrid
Nombre del trabajo especificado por usuario: SumaVectoresC_vdinamicos
Modo que ejecuta qsub: atcgrid
Directorio en el que se ha ejecutado qsub: /home/Blestudiante3/hello
Cola: ac
Nodos asignados al trabajo:
atcgrid1
Tiempo(seg.):0.000433693 / 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) /
Tiempo(seg.):0.000910943 / Tamaño Vectores:131072 / V1[0]+V2[0]=V3[0](13107.200000+13107.200000=26214.400000) / V1[131071]+V2[131071]=V3[131071](26214.30
0000+0.100000=26214.400000) /
Tiempo(seg.):0.001776552 / Tamaño Vectores:262144 / V1[0]+V2[0]=V3[0](26214.400000+26214.400000=52428.800000) / V1[262143]+V2[262143]=V3[262143](52428.70
0000+0.100000=52428.800000) /
Tiempo(seg.):0.002713073 / 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) /
Tiempo(seg.):0.004446811 / Tamaño Vectores:1048576 / V1[0]+V2[0]=V3[0](104857.600000+104857.600000=209715.200000) / V1[1048575]+V2[1048575]=V3[1048575](20
9715.100000+0.100000=209715.200000) /
Tiempo(seg.):0.008416911 / Tamaño Vectores:2097152 / V1[0]+V2[0]=V3[0](209715.200000+209715.200000=419430.400000) / V1[2097151]+V2[2097151]=V3[2097151](41
9430.300000+0.100000=419430.400000) /
Tiempo(seg.):0.016176787 / Tamaño Vectores:4194304 / V1[0]+V2[0]=V3[0](419430.400000+419430.400000=838860.800000) / V1[4194303]+V2[4194303]=V3[4194303](83
8860.700000+0.100000=838860.800000) /
Tiempo(seg.):0.031653135 / Tamaño Vectores:8388608 / V1[0]+V2[0]=V3[0](838860.800000+838860.800000=1677721.600000) / V1[8388607]+V2[8388607]=V3[8388607](1
677721.500000+0.100000=1677721.600000) /
Tiempo(seg.):0.062658369 / Tamaño Vectores:16777216 / V1[0]+V2[0]=V3[0](1677721.600000+1677721.600000=3355443.200000) / V1[16777215]+V2[16777215]=V3[167772
15](3355443.100000+0.100000=3355443.200000) /
Tiempo(seg.):0.124663532 / Tamaño Vectores:33554432 / V1[0]+V2[0]=V3[0](3355443.200000+3355443.200000=6710886.400000) / V1[33554431]+V2[33554431]=V3[335544
31](6710886.300000+0.100000=6710886.400000) /
Tiempo(seg.):0.248541566 / Tamaño Vectores:67108864 / V1[0]+V2[0]=V3[0](6710886.400000+6710886.400000=13421772.800000) / V1[67108863]+V2[67108863]=V3[671088
63](13421772.700000+0.100000=13421772.800000) /
```

## Por mi ordenador:

```
[Blestudiante3@atcgrid hello]$ cat SumaVectoresC_vglobales_mipc.txt
Id. usuario del trabajo: Blestudiante3
Id. del trabajo: 25312.atcgrid
Nombre del trabajo especificado por usuario: SumaVectoresC_vglobales_mipc
Modo que ejecuta qsub: atcgrid
Directorio en el que se ha ejecutado qsub: /home/Blestudiante3/hello
Cola: ac
Nodos asignados al trabajo:
atcgrid1
Tiempo(seg.):0.000750732 / 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) /
Tiempo(seg.):0.001541057 / Tamaño Vectores:131072 / V1[0]+V2[0]=V3[0](13107.200000+13107.200000=26214.400000) / V1[131071]+V2[131071]=V3[131071](26214.30
0000+0.100000=26214.400000) /
Tiempo(seg.):0.001853131 / Tamaño Vectores:262144 / V1[0]+V2[0]=V3[0](26214.400000+26214.400000=52428.800000) / V1[262143]+V2[262143]=V3[262143](52428.70
0000+0.100000=52428.800000) /
Tiempo(seg.):0.004389781 / 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) /
Tiempo(seg.):0.009178964 / Tamaño Vectores:1048576 / V1[0]+V2[0]=V3[0](104857.600000+104857.600000=209715.200000) / V1[1048575]+V2[1048575]=V3[1048575](20
9715.100000+0.100000=209715.200000) /
Tiempo(seg.):0.016314290 / Tamaño Vectores:2097152 / V1[0]+V2[0]=V3[0](209715.200000+209715.200000=419430.400000) / V1[2097151]+V2[2097151]=V3[2097151](41
9430.300000+0.100000=419430.400000) /
Tiempo(seg.):0.032150456 / Tamaño Vectores:4194304 / V1[0]+V2[0]=V3[0](419430.400000+419430.400000=838860.800000) / V1[4194303]+V2[4194303]=V3[4194303](83
8860.700000+0.100000=838860.800000) /
Tiempo(seg.):0.054805346 / Tamaño Vectores:8388608 / V1[0]+V2[0]=V3[0](838860.800000+838860.800000=1677721.600000) / V1[8388607]+V2[8388607]=V3[8388607](1
677721.500000+0.100000=1677721.600000) /
Tiempo(seg.):0.120323727 / Tamaño Vectores:16777216 / V1[0]+V2[0]=V3[0](1677721.600000+1677721.600000=3355443.200000) / V1[16777215]+V2[16777215]=V3[167772
15](3355443.100000+0.100000=3355443.200000) /
Tiempo(seg.):0.221183270 / Tamaño Vectores:33554432 / V1[0]+V2[0]=V3[0](3355443.200000+3355443.200000=6710886.400000) / V1[33554431]+V2[33554431]=V3[335544
31](6710886.300000+0.100000=6710886.400000) /
Tiempo(seg.):0.215990839 / Tamaño Vectores:33554432 / V1[0]+V2[0]=V3[0](3355443.200000+3355443.200000=6710886.400000) / V1[33554431]+V2[33554431]=V3[335544
31](6710886.300000+0.100000=6710886.400000) /
```



```

fbilupi@fbilupi-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC/Practicas/P05 cat SumaVectoresC_vdinamicos_mipc.txt
Tiempo(seg.):0.000746323 / 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) /
Tiempo(seg.):0.001560532 / Tamaño Vectores:131072 / V1[0]+V2[0]=V3[0](13107.200000+13107.200000=26214.400000) // V1[131071]+V2[131071]=V3[131071](26214.30
0000+0.100000=26214.400000) /
Tiempo(seg.):0.002444211 / Tamaño Vectores:262144 / V1[0]+V2[0]=V3[0](26214.400000+26214.400000=52428.800000) // V1[262143]+V2[262143]=V3[262143](52428.70
0000+0.100000=52428.800000) /
Tiempo(seg.):0.004291428 / 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) /
Tiempo(seg.):0.007219076 / Tamaño Vectores:1048576 / V1[0]+V2[0]=V3[0](104857.600000+104857.600000=209715.200000) // V1[1048575]+V2[1048575]=V3[1048575](20
9715.100000+0.100000=209715.200000) /
Tiempo(seg.):0.013885750 / Tamaño Vectores:2097152 / V1[0]+V2[0]=V3[0](209715.200000+209715.200000=419430.400000) // V1[2097151]+V2[2097151]=V3[2097151](41
9430.300000+0.100000=419430.400000) /
Tiempo(seg.):0.027906939 / Tamaño Vectores:4194304 / V1[0]+V2[0]=V3[0](419430.400000+419430.400000=838860.800000) // V1[4194303]+V2[4194303]=V3[4194303](83
8860.700000+0.100000=838860.800000) /
Tiempo(seg.):0.054446909 / Tamaño Vectores:8388608 / V1[0]+V2[0]=V3[0](838860.800000+838860.800000=1677721.600000) // V1[8388607]+V2[8388607]=V3[8388607](1
677721.500000+0.100000=1677721.600000) /
Tiempo(seg.):0.108429232 / Tamaño Vectores:16777216 / V1[0]+V2[0]=V3[0](1677721.600000+1677721.600000=3355443.200000) // V1[16777215]+V2[16777215]=V3[167772
15](3355443.100000+0.100000=3355443.200000) /
Tiempo(seg.):0.217058945 / Tamaño Vectores:33554432 / V1[0]+V2[0]=V3[0](3355443.200000+3355443.200000=6710886.400000) // V1[33554431]+V2[33554431]=V3[335544
31](6710886.300000+0.100000=6710886.400000) /
Tiempo(seg.):0.433402721 / Tamaño Vectores:67108864 / V1[0]+V2[0]=V3[0](6710886.400000+6710886.400000=13421772.800000) // V1[67108863]+V2[67108863]=V3[67108
863](13421772.700000+0.100000=13421772.800000) /

```

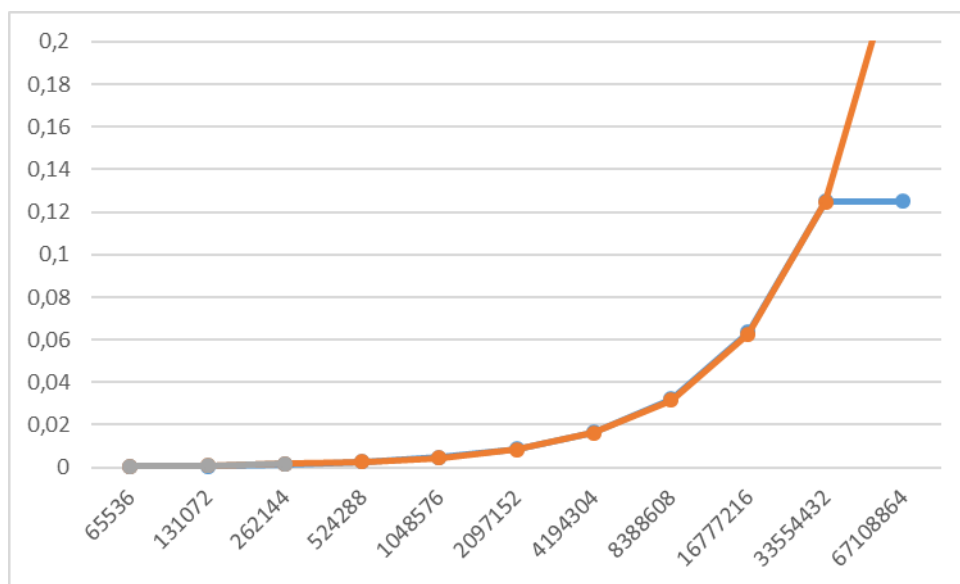
Como vemos, nunca se obtiene error. Porque en el caso del vector global, las variables son globales y su longitud no estará limitada por el tamaño de la pila del programa. En el caso del vector dinámico se debe a que este tipo de vector reutiliza la memoria durante la ejecución.

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). Realice otra gráfica con los tiempos obtenidos en el PC local.

**RESPUESTA:**

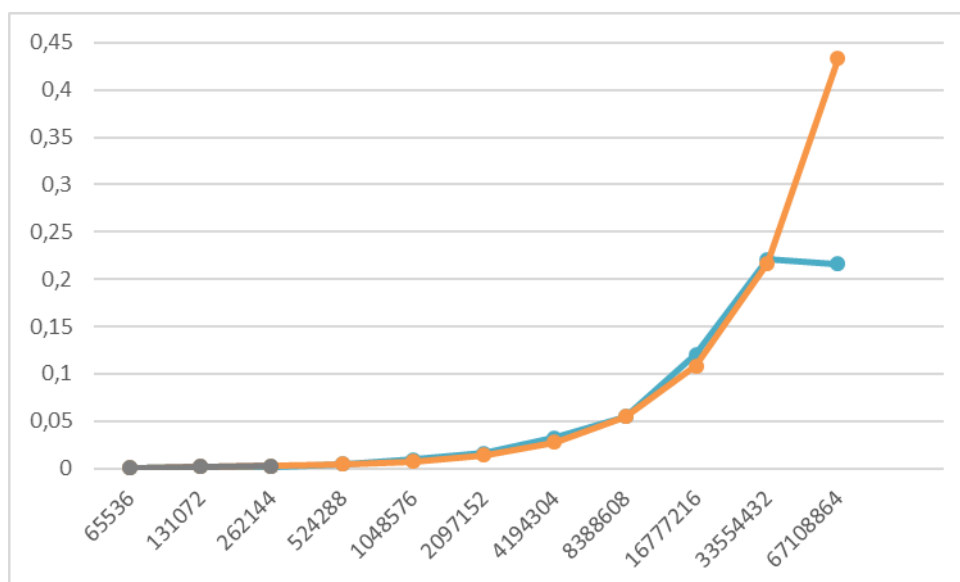
**Tabla 1 .** Tiempos de ejecución de la suma de vectores para vectores locales, globales y dinámicos

Nº de Componentes	Bytes de un vector	Tiempo para vect. locales	Tiempo para vect. globales	Tiempo para vect. dinámicos
65536	524288	0.000438491	0.000549985	0.000433693
131072	1048576	0.000882654	0.000534678	0.000910943
262144	2097152	0.001784521	0.001436022	0.001778652
524288	4194304	-	0.002640176	0.002713073
1048576	8388608	-	0.004801694	0.004446811
2097152	16777216	-	0.008574328	0.008416911
4194304	33554432	-	0.016399286	0.016176787
8388608	67108864	-	0.032141182	0.031655135
16777216	134217728	-	0.063612018	0.062658369
33554432	268435456	-	0.124973505	0.124663532
67108864	536870912	-	0.124966156	0.248541566



Vector local, vector global y vector dinámico.

Vemos en esta gráfica con los datos de los tiempos de ATCGRID que los tiempos son muy similares.



Vector local, vector global y vector dinámico.

En el caso de los tiempos en mi pc, El dibujo que hacen las gráficas sigue siendo similar, pero con unos valores más altos.

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? Razone además por qué el máximo número que se puede almacenar en N es  $2^{32}-1$ .

#### RESPUESTA:

Nos aparece el siguiente error:

```

falupiefbiupl-ElementaryOS:~/Dropbox/FACULTAD/Grado Informatica/2o Curso/2o Cuatrimestre/AC/Practicas/P0$ gcc -O2 SumaVectoresC.c -o SumaVectoresGlobal232 -lrt
/tmp/ccSugQMa.o: In function 'main':
SumaVectoresC.c:(.text.startup+0x66): relocation truncated to fit: R_X86_64_32S against symbol `v2' defined in COMMON section in /tmp/ccSugQMa.o
SumaVectoresC.c:(.text.startup+0x8e): relocation truncated to fit: R_X86_64_32S against symbol `v2' defined in COMMON section in /tmp/ccSugQMa.o
SumaVectoresC.c:(.text.startup+0x97): relocation truncated to fit: R_X86_64_32S against symbol `v3' defined in COMMON section in /tmp/ccSugQMa.o
SumaVectoresC.c:(.text.startup+0xb7): relocation truncated to fit: R_X86_64_PC32 against symbol `v3' defined in COMMON section in /tmp/ccSugQMa.o
SumaVectoresC.c:(.text.startup+0xce): relocation truncated to fit: R_X86_64_PC32 against symbol `v2' defined in COMMON section in /tmp/ccSugQMa.o
SumaVectoresC.c:(.text.startup+0xd7): relocation truncated to fit: R_X86_64_32S against symbol `v3' defined in COMMON section in /tmp/ccSugQMa.o
SumaVectoresC.c:(.text.startup+0xf2): relocation truncated to fit: R_X86_64_32S against symbol `v2' defined in COMMON section in /tmp/ccSugQMa.o
collect2: ld devolvió el estado de salida 1

```



- 11.** Describir las diferencias, ventajas e inconvenientes en la utilización de vectores y matrices locales, globales y dinámicos.

**RESPUESTA:** El problema que tenemos con los vectores locales es que si se supera el tamaño de la pila se producirá una violación de segmento, aunque su tiempo es menor comparado con los vectores dinámicos (en caso de no fallar por superar el tamaño de la pila). Los vectores dinámicos, en cambio, no tendrán el mismo problema que los estáticos aunque son un poco más lentos. El caso de los vectores globales es similar al de los dinámicos pero con el inconveniente de que pueden estar ocupando mucha memoria que no les hace falta en un momento determinado.