

Arquitectura de Computadores (AC)

Cuaderno de prácticas.

Bloque Práctico 0. Entorno de programación

Estudiante (nombre y apellidos): Ignacio Morillas Padial

Grupo de prácticas: A3

Fecha de entrega: 08/03/2018

Fecha evaluación en clase: 09/03/2018

1. Incorpore volcados de pantalla que muestren lo que devuelve `lscpu` en `atcgrid` y en su PC.

CAPTURAS:

Mi PC:

```
ignacio@ignacio-PC:~/universidad/AC/P0$ PS1="[IgnacioMorillasPadial \u@h:w] \D{%F %A}\n$"  
[IgnacioMorillasPadial ignacio@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves  
$lscpu  
Arquitectura: x86_64  
modo(s) de operación de las CPUs: 32-bit, 64-bit  
Orden de bytes: Little Endian  
CPU(s): 4  
On-line CPU(s) list: 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: 61  
Model name: Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz  
Revisión: 4  
CPU MHz: 799.890  
CPU max MHz: 2700.0000  
CPU min MHz: 500.0000  
BogoMIPS: 4393.52  
Virtualización: VT-x  
Caché L1d: 32K  
Caché L1i: 32K  
Caché L2: 256K  
Caché L3: 3072K  
NUMA node0 CPU(s): 0-3  
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 sdbg fma cx16 xtpr pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand lahf_lm abm 3  
dnowprefetch epb intel_pt tpr_shadow vnmi flexpriority ept vpid fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap xsaveopt d  
therm ida arat pln pts  
[IgnacioMorillasPadial ignacio@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves  
$
```

Atcgrid:

```
[A3estudiante16@atcgrid ~]$ cat STDIN.065762  
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: 1600.102  
CPU max MHz: 2401.0000  
CPU min MHz: 1600.0000  
BogoMIPS: 4800.14  
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 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe sy  
scall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf pni dtes64 monitor ds_cpl vm  
x smx est tm2 ssse3 cx16 xtpr pdcm pcid dca sse4_1 sse4_2 popcnt lahf_lm epb pti retpoline tpr_shadow vnmi flexpriority ept vpid dtherm ida ara  
t  
[A3estudiante16@atcgrid ~]$
```

1. Conteste a las siguientes preguntas:

- a. ¿Cuántos cores físicos y cuántos cores lógicos tiene atcgrid de prácticas o su PC?

RESPUESTA: En mi PC tengo 2 cores físicos, y por tanto, 4 cores lógicos

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

RESPUESTA:

En el nodo de atcgrid tiene 12 cores físicos, y por tanto, 24 cores lógicos.

2. 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`.

- 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` es un `double` que contiene el tiempo que tarda en ejecutarse la suma de los vectores.

`clock_gettime` devuelve un 0 o un -1, si ha tenido éxito la función, lo que nos interesa es el segundo argumento que se le pasa a dicha función, un puntero de `timespec`, este parámetro es un `struct` que contiene 2 atributos:

`-time_t tv_sec` → Devuelve los segundos que han pasado desde 1970

`-time_t tv_nsec` → Devuelve los nanosegundos que han pasado del segundo actual.

El tiempo transcurrido entre el principio y el final de la ejecución de la suma de vectores se calcula restando al `timespec` final el inicial.

- 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++
bibliotecas	<code>#include<stdlib.h></code> <code>#include<stdio.h></code>	<code>#include<cstdlib.h></code> <code>#include<isostream></code>
Reserva de memoria	<code>v1=(double*)</code> <code>malloc(N*sizeof(double));</code>	<code>v1=new double[N];</code>
Salida de datos por pantalla	<code>printf("tiempo(seg):</code> <code>%11.9f\t / Tamaño Vectores:</code> <code>%u\n",ncgt,N);</code>	<code>cout<<"tiempo(seg):"<</code> <code>ncgt << "\t/ Tamaño</code> <code>Vectores:"<<N<<endl;</code>
Liberar memoria	<code>free(v1);</code>	<code>delete[]v1;</code>

3. 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). Incorporar volcados de pantalla que demuestren la ejecución correcta en atcgrid o en su PC.

RESPUESTA:

PC:

```
Sgcc -fopenmp -o sumavectoresdinamic sumavectoresC.c
[IgnacioMorilasPadial ignacio@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves
$ ./scriptc.sh
Tiempo(seg.):0.000563847 / 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.):0.000654617 / 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) /

Tiempo(seg.):0.001272242 / 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) /

./scriptc.sh: línea 7: 6961 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6963 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6965 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6967 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6969 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6971 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6973 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6975 Violación de segmento ('core' generado) ./sumavectoreslocal $N
```

Atcgrid:

```
[IgnacioMorilasPadi@ignacio@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves
$ ./scriptc.sh
Tiempo(seg.):0.000563847 / 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.):0.000654617 / 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) /

Tiempo(seg.):0.001272242 / 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) /

./scriptc.sh: línea 7: 6961 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6963 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6965 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6967 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6969 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6971 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6973 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 6975 Violación de segmento ('core' generado) ./sumavectoreslocal $N
```

4. 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 para los mismos tamaños. ¿Se obtiene error para alguno de los tamaños? En caso afirmativo, ¿a qué se debe este error? (Incorporar volcados de pantalla)

RESPUESTA:

PC(local)(O2):

```
A3estudiante16@atcgriid -j$ cat STDIN.o65781
Id. usuario del trabajo: A3estudiante16
Id. del trabajo: 65781.atcgriid
Nombre del trabajo especificado por usuario: STDIN
Nodo que ejecuta qsub: atcgriid
Directorio en el que se ha ejecutado qsub: /home/A3estudiante16
Cola: ac
Nodos asignados al trabajo:
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
atcgriid1
Tiempo(seg.):0.000420716 / 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.):0.000851922 / 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) /
Tiempo(seg.):0.001114684 / 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) /
```

Atcgrid(local)O2:

```

[IgnacioMorilasPadial ignacio@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves
$ ./scriptc.sh
Tiempo(seg.):0.000265295 / 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.):0.000733360 / 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) /

Tiempo(seg.):0.001370395 / 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) /

./scriptc.sh: línea 7: 4421 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 4423 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 4425 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 4427 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 4429 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 4431 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 4433 Violación de segmento ('core' generado) ./sumavectoreslocal $N
./scriptc.sh: línea 7: 4435 Violación de segmento ('core' generado) ./sumavectoreslocal $N
[IgnacioMorilasPadial ignacio@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves

```

5. 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. ¿Se obtiene error usando vectores globales o dinámicos? ¿A qué cree que es debido? (Incorporar volcados de pantalla)

RESPUESTA:

PC(Global):

```

[IgnacioMorilasPadial ignacio@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves
$ ./scriptc.sh
Tiempo(seg.):0.000518797 / 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.):0.001018911 / 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) /

Tiempo(seg.):0.001340879 / 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) /

Tiempo(seg.):0.002169437 / 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.):0.005810814 / 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) /

Tiempo(seg.):0.008904689 / 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) /

Tiempo(seg.):0.017680404 / 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) /

Tiempo(seg.):0.034119198 / 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) /

Tiempo(seg.):0.065539947 / 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) /

Tiempo(seg.):0.130984218 / 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) /

Tiempo(seg.):0.358672943 / 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) /

```


Atcgrid(Global):

```
[A3estudiante16@atcgrid ~]$ cat STDIN.o65798
Tiempo(seg.):0.000430029          / 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.):0.000924703          / 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) /

Tiempo(seg.):0.001710685          / 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) /

Tiempo(seg.):0.003886912          / 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.):0.005785725          / 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) /

Tiempo(seg.):0.011778109          / 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) /

Tiempo(seg.):0.022907012          / 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) /

Tiempo(seg.):0.046451050          / 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) /

Tiempo(seg.):0.100230307          / 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) /

Tiempo(seg.):0.184994112          / 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) /

Tiempo(seg.):0.353044502          / 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) /
```

PC(Dinamico):

```

[IgnacioMorilasPadial ignacio@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves
$./scriptc.sh
Tiempo(seg.):0.000615761 / 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.):0.000645247 / 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) /

Tiempo(seg.):0.002074267 / 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) /

Tiempo(seg.):0.002050657 / 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.):0.008971197 / 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) /

Tiempo(seg.):0.010031124 / 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) /

Tiempo(seg.):0.025720526 / 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) /

Tiempo(seg.):0.034353630 / 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) /

Tiempo(seg.):0.067756863 / 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) /

Tiempo(seg.):0.138277289 / 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) /

Tiempo(seg.):0.276733994 / 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) /

[IgnacioMorilasPadial ignacio@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves

```

Arcgrid(Dinamico):

```

[IgnacioMorilasPadi@ignacio-PC:~/universidad/AC/P0] 2018-03-08 jueves
$ ./scriptc.sh
Tiempo(seg.):0.000615761 / 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.):0.000645247 / 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) /

Tiempo(seg.):0.002074267 / 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) /

Tiempo(seg.):0.002050657 / 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.):0.008971197 / 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) /

Tiempo(seg.):0.010031124 / 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) /

Tiempo(seg.):0.025720526 / 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) /

Tiempo(seg.):0.034353630 / 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) /

Tiempo(seg.):0.067756863 / 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) /

Tiempo(seg.):0.138277289 / 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) /

Tiempo(seg.):0.276733994 / 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) /

```

6. Rellenar una tabla como la Tabla 1 para atcgrid y otra para su PC 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 y en su PC para vectores locales, globales y dinámicos (eje y) en función del tamaño en bytes de un vector (los valores de la segunda columna de la tabla, que están en escala logarítmica, deben estar en el eje x). Utilice escala logarítmica en el eje de ordenadas (eje y). ¿Hay diferencias en los tiempos de ejecución?

RESPUESTA:

Tabla 1 . Mi PC

Nº de Componentes	Bytes de un vector	Tiempo para vect. locales	Tiempo para vect. globales	Tiempo para vect. dinámicos
65536	524288	0,000265295	0,000518797	0,000615761
131072	1048576	0,000733360	0,001018911	0,000645247
262144	2097152	0,001370395	0,001340879	0,002074267
524288	4194304		0,002169437	0,002050657
1048576	8388608		0,005810814	0,008971197
2097152	16777216		0,008904689	0,010031124
4194304	33554432		0,017680404	0,025720526
8388608	67108864		0,034119198	0,034353630
16777216	134217728		0,065539947	0,067756863
33554432	268435456		0,130984218	0,138277289
67108864	536870912		0,358672943	<u>0,276733994</u>

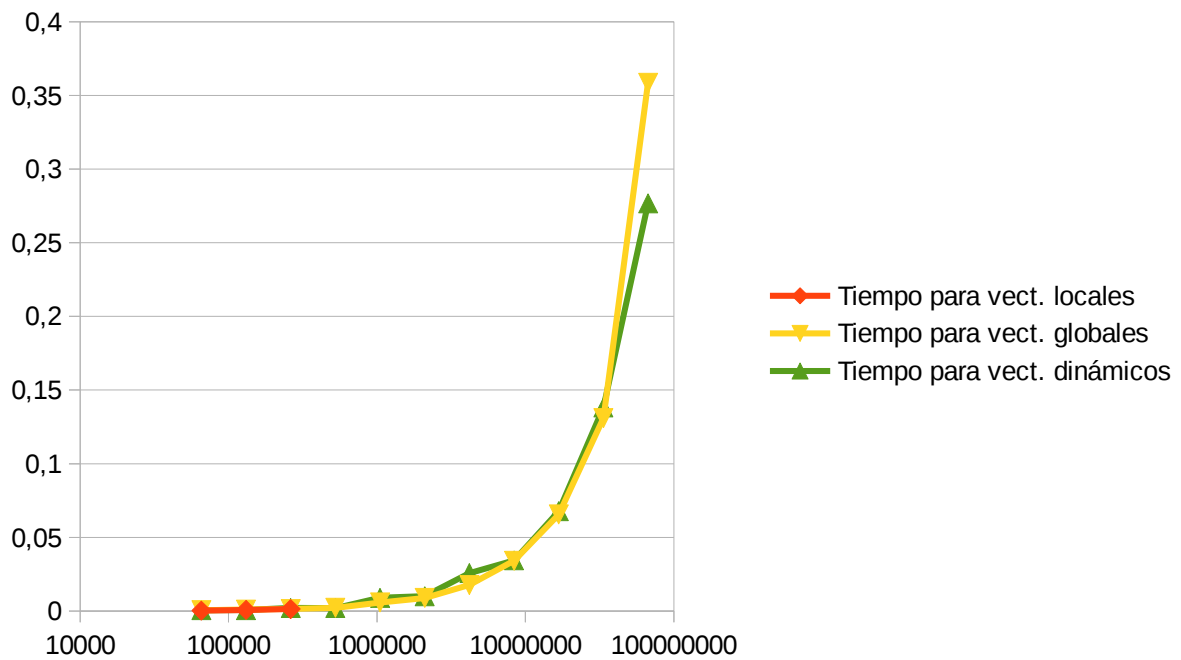
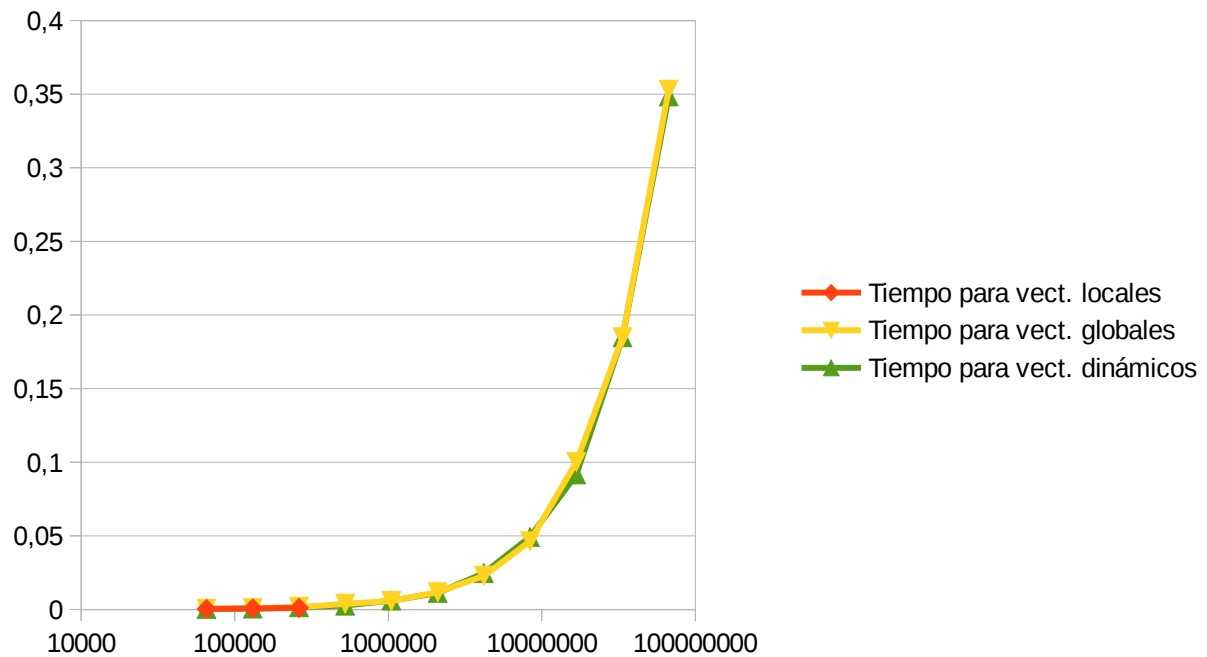


Tabla 2 .Atcgrid

Nº de Componentes	Bytes de un vector	Tiempo para vect. locales	Tiempo para vect. globales	Tiempo para vect. dinámicos
65536	524288	0.000420716	0.000430029	0.000418984
131072	1048576	0.000851922	0.000924703	0.000593868
262144	2097152	0.001114684	0.001710685	0.001521966
524288	4194304		0.003886912	0.002766557
1048576	8388608		0.005785725	0.006108354
2097152	16777216		0.011778109	0.011341413
4194304	33554432		0.022907012	0.024787773
8388608	67108864		0.046451050	0.049260704
16777216	134217728		0.100230307	0.091768434
33554432	268435456		0.184994112	0.184896561
67108864	536870912		0.353044502	0.348478029



7. 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 ($\text{MAX}=2^{32}-1$). 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:

```
ignacio@ignacio-PC:~/universidad/AC/P05$ gcc -lrt -O2 -o prueba sumavectoresC.c
/tmp/ccsYVPvV.o: En la función 'main':
sumavectoresC.c:(.text.startup+0x79): reubicación truncada para ajustar: R_X86_64_32S contra el símbolo 'v2' definido en la sección COMMON en /tmp/ccsYVPvV.o
sumavectoresC.c:(.text.startup+0xc0): reubicación truncada para ajustar: R_X86_64_32S contra el símbolo 'v2' definido en la sección COMMON en /tmp/ccsYVPvV.o
sumavectoresC.c:(.text.startup+0xc8): reubicación truncada para ajustar: R_X86_64_32S contra el símbolo 'v3' definido en la sección COMMON en /tmp/ccsYVPvV.o
sumavectoresC.c:(.text.startup+0xfc): reubicación truncada para ajustar: R_X86_64_32S contra el símbolo 'v3' definido en la sección COMMON en /tmp/ccsYVPvV.o
sumavectoresC.c:(.text.startup+0x115): reubicación truncada para ajustar: R_X86_64_32S contra el símbolo 'v2' definido en la sección COMMON en /tmp/ccsYVPvV.o
sumavectoresC.c:(.text.startup+0x12b): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo 'v3' definido en la sección COMMON en /tmp/ccsYVPvV.o
sumavectoresC.c:(.text.startup+0x135): reubicación truncada para ajustar: R_X86_64_PC32 contra el símbolo 'v2' definido en la sección COMMON en /tmp/ccsYVPvV.o
collect2: error: ld returned 1 exit status
ignacio@ignacio-PC:~/universidad/AC/P05$
```

El número máximo que puede almacenar N es $2^{32}-1$ ya que N es un unsigned int, el número máximo que soporta es $2^{32}-1$ (registro de bits 32 por tanto : 11111....)

Ha ocurrido un error a la hora de compilar el programa cuando se cambia el valor máximo a $2^{32}-1$ ya que se supera el máximo de elementos que soporta un vector.