

## 1.1

2)

a)

Genero el ejecutable con optimización -O0

```
$ gcc -O0 pi.c -o pi.O0
```

b)

Ejecuto el ejecutable pi.O0 y guardo el resultado en pi.0\_time

```
$ /usr/bin/time -o pi.0_time ./pi.O0
```

```
$ cat pi.0_time
```

```
55.69user 0.18system 0:55.89elapsed 99%CPU (0avgtext+0avgdata
852maxresident)k 0inputs+0outputs (0major+64minor)pagefaults 0swaps
```

c)

El %CPU es el tanto por ciento de CPU usada para ejecutar el programa

3)

a)

Genero los ejecutables correspondientes

```
$ gcc -O1 pi.c -o pi.O1
```

```
$ gcc -O2 pi.c -o pi.O2
```

```
$ gcc -O3 pi.c -o pi.O3
```

Ejecuto el ejecutable pi.O3 y guardo el resultado en pi.3\_time

```
$/usr/bin/time -o pi.3_time ./pi.O3
```

```
$ cat pi.3_time
```

```
17.80user 0.16system 0:17.99elapsed 99%CPU (0avgtext+0avgdata      984maxresident)k
0inputs+0outputs (0major+66minor)pagefaults 0swaps
```

Vemos que el resultado es diferente en el tiempo usado por el usuario y el sistema, y por lo tanto, en el elapsed time. Esto es debido a que ha habido unas optimizaciones en el código y ahora se ejecuta más rápido

b) i c)

He añadido un tipo de formato diferente para ver el resultado como quiero para evaluar las diferencias mejor

```
$ /usr/bin/time -o pi.0_time -f "Elapsed time: %e User: %U System: %S " ./pi.O0
```

```
$ /usr/bin/time -o pi.3_time -f "Elapsed time: %e User: %U System: %S " ./pi.O3
```

## 1.2

4)

a) i b)

Vemos que la rutina más invocada por el programa es "SUBTRACT" con 2011 calls. Y el mayor tiempo de CPU consumido por una rutina del programa es 26.32% (supongo que \_\_aeabi\_uidiv no es una rutina del programa)

```
$ gcc -pg -g -O0 pi.c -o pi.O0.pg #Compilamos con -g y -pg para poder utilizar gprof
```

```
$ ./pi.O0.pg 1000 #Y ejecutamos antes de gprof siempre
```

```
$ gprof -b ./pi.O0.pg > pi.O0.pg.gprof.txt
```

```
$ cat pi.O0.pg.gprof.txt
```

Flat profile:

Each sample counts as 0.01 seconds.

	% cumulative	self		self	total	
time	seconds	seconds	calls	ms/call	ms/call	name
47.37	0.27	0.27				__aeabi_uidiv
26.32	0.42	0.15	2011	0.07	0.07	SUBTRACT
24.56	0.56	0.14	3014	0.05	0.05	DIVIDE
1.75	0.57	0.01	1004	0.01	0.01	LONGDIV
0.00	0.57	0.00	1007	0.00	0.00	SET
0.00	0.57	0.00	1005	0.00	0.00	progress
0.00	0.57	0.00	2	0.00	0.00	MULTIPLY
0.00	0.57	0.00	1	0.00	300.00	calculate
0.00	0.57	0.00	1	0.00	0.00	epilog

c)

Ejecutamos gprof con el flag -l para ver la información por línea de código. Por línea de código observamos que DIVIDE (línea 17 del programa) consume el mayor tiempo de CPU con 7.89 %

```
$ gprof -b -l ./pi.O0.pg > pi.O0.pg.gprof.txt
```

Each sample counts as 0.01 seconds.

	% cumulative	self		self	total	
time	seconds	seconds	calls	Ts/call	Ts/call	name
47.37	0.27	0.27				__aeabi_uidiv
7.89	0.32	0.04				DIVIDE (pi.c:17 @ 10792)
7.89	0.36	0.04				SUBTRACT (pi.c:91 @ 109e8)
7.02	0.40	0.04				DIVIDE (pi.c:19 @ 107b6)
7.02	0.44	0.04				SUBTRACT (pi.c:94 @ 10a30)
6.14	0.47	0.04				SUBTRACT (pi.c:93 @ 10a16)
5.26	0.51	0.03				SUBTRACT (pi.c:89 @ 10a46)
3.51	0.53	0.02				DIVIDE (pi.c:18 @ 107a8)
3.51	0.55	0.02				DIVIDE (pi.c:15 @ 107d0)
1.75	0.56	0.01				DIVIDE (pi.c:20 @ 107c4)
1.75	0.56	0.01				LONGDIV (pi.c:37 @ 10838)
0.88	0.57	0.01				DIVIDE (pi.c:15 @ 1078c)
0.00	0.57	0.00	3014	0.00	0.00	DIVIDE (pi.c:9 @ 10778)
0.00	0.57	0.00	2011	0.00	0.00	SUBTRACT (pi.c:85 @ 109c8)
0.00	0.57	0.00	1007	0.00	0.00	SET (pi.c:78 @ 10990)
0.00	0.57	0.00	1005	0.00	0.00	progress (pi.c:181 @ 10c8c)
0.00	0.57	0.00	1004	0.00	0.00	LONGDIV (pi.c:25 @ 107f0)
0.00	0.57	0.00	2	0.00	0.00	MULTIPLY (pi.c:64 @ 1090c)
0.00	0.57	0.00	1	0.00	0.00	calculate (pi.c:126 @ 10b08)
0.00	0.57	0.00	1	0.00	0.00	epilog (pi.c:186 @ 10ca0)

d)

No, en gprof no muestra nada del sistema igual que en Valgrind

5)

a) i b)

```
$ gcc -pg -g -O3 pi.c -o pi.O3.pg
```

```
$ ./pi.O3.pg 1000
$ gprof -b ./pi.O3.pg > pi.O3.pg.gprof.txt
$ cat pi.O3.pg.gprof.txt
```

Solo aparece la funcion calculate, debido seguramente a que copia el código de todas las funciones a calculate que es donde se llaman, mejorando el tiempo de ejecución pues se hacen menos calls (inline). Ello conlleva tambien a que aumente el peso del ejecutable. Pues si hay dos llamadas de una función, el código de esta se ve duplicado.

time	seconds	seconds	calls	Ts/call	Ts/call	name
70.66	0.12	0.12				calculate
29.44	0.17	0.05				__aeabi_uidiv

He mirado por línea de código a ver que pasaba, y como se aprecia, no tienen ninguna llamada, por lo que reafirmo lo que dije anteriormente (inline de las funciones)

```
$ gprof -b -l ./pi.O3.pg > pi.O3.pg.gprof.txt
```

time	seconds	seconds	calls	Ts/call	Ts/call	name
29.44	0.05	0.05				__aeabi_uidiv
11.78	0.07	0.02				SUBTRACT (pi.c:91 @ 10ada)
8.83	0.09	0.02				SUBTRACT (pi.c:94 @ 10b80)
5.89	0.10	0.01				DIVIDE (pi.c:18 @ 10bd6)
5.89	0.11	0.01				DIVIDE (pi.c:20 @ 10bdc)
5.89	0.12	0.01				SUBTRACT (pi.c:93 @ 10aee)
5.89	0.13	0.01				SUBTRACT (pi.c:89 @ 10afa)
5.89	0.14	0.01				SUBTRACT (pi.c:91 @ 10b68)
2.94	0.14	0.01				DIVIDE (pi.c:18 @ 10b46)
2.94	0.15	0.01				DIVIDE (pi.c:20 @ 10b4c)
2.94	0.15	0.01				LONGDIV (pi.c:33 @ 10c1c)
2.94	0.16	0.01				LONGDIV (pi.c:35 @ 10c34)
2.94	0.16	0.01				LONGDIV (pi.c:37 @ 10c4c)
2.94	0.17	0.01				LONGDIV (pi.c:33 @ 10c50)
2.94	0.17	0.01				SUBTRACT (pi.c:89 @ 10b88)

6)

a)

Las diferencias en la columna de samples es debido a que oprofile utiliza el evento predeterminado para Linux llamado CYCLES, que cuenta los ciclos del procesador. La diferencia de este contador determina la frecuencia con la que ocurre un evento y por lo tanto, esto afecta al recuento de muestras de cada evento. De ahí las diferencias de recuento de las muestras (samples)

```
$ gcc -g -O0 pi.c -o pi.O0.g
$ operf --event=CPU_CYCLES:100000 ./pi.O0.g
$ opreport -l
```

CPU: ARM Cortex-A9, speed 999 MHz (estimated)

Counted CPU\_CYCLES events (CPU cycle) with a unit mask of 0x00 (No unit mask) count 100000

samples	%	image name	symbol name
130235	33.5958	pi.O0.g	__aeabi_uidiv
109775	28.3179	pi.O0.g	SUBTRACT

94519	24.3824	pi.O0.g	DIVIDE
34861	8.9928	pi.O0.g	LONGDIV
13832	3.5681	pi.O0.g	.divsi3_skip_div0_test
3594	0.9271	no-vmlinux	/no-vmlinux
289	0.0746	pi.O0.g	__divsi3
199	0.0513	libc-2.23.so	memset
67	0.0173	libc-2.23.so	vfprintf
39	0.0101	libc-2.23.so	_IO_file_write@@GLIBC_2.4
34	0.0088	libc-2.23.so	new_do_write
31	0.0080	libc-2.23.so	putchar
22	0.0057	libc-2.23.so	write
22	0.0057	pi.O0.g	calculate
16	0.0041	libc-2.23.so	_IO_file_overflow@@GLIBC_2.4
16	0.0041	libc-2.23.so	_IO_file_xsputn@@GLIBC_2.4
16	0.0041	pi.O0.g	epilog
14	0.0036	libc-2.23.so	buffered_vfprintf
11	0.0028	libc-2.23.so	strchrnul
10	0.0026	libc-2.23.so	_IO_do_write@@GLIBC_2.4
10	0.0026	pi.O0.g	MULTIPLY
8	0.0021	libc-2.23.so	_IO_default_xsputn
7	0.0018	pi.O0.g	SET
6	0.0015	libc-2.23.so	__overflow
6	0.0015	libc-2.23.so	_itoa_word
5	0.0013	pi.O0.g	progress
4	0.0010	libc-2.23.so	fputc
1	2.6e-04	ld-2.23.so	_dl_relocate_object
1	2.6e-04	ld-2.23.so	check_match
1	2.6e-04	ld-2.23.so	open
1	2.6e-04	libc-2.23.so	_IO_flush_all_lockp
1	2.6e-04	libc-2.23.so	fwrite

```
$ perf --event=CPU_CYCLES:750000 ./pi.O0.g
$ opreport -l
```

CPU: ARM Cortex-A9, speed 999 MHz (estimated)

Counted CPU\_CYCLES events (CPU cycle) with a unit mask of 0x00 (No unit mask) count 750000

samples	%	image name	symbol name
16731	33.2955	pi.O0.g	__aeabi_uidiv
14284	28.4259	pi.O0.g	SUBTRACT
12407	24.6905	pi.O0.g	DIVIDE
4470	8.8955	pi.O0.g	LONGDIV
1775	3.5323	pi.O0.g	.divsi3_skip_div0_test
467	0.9294	no-vmlinux	/no-vmlinux
41	0.0816	pi.O0.g	__divsi3
26	0.0517	libc-2.23.so	memset
11	0.0219	libc-2.23.so	write
8	0.0159	libc-2.23.so	vfprintf
4	0.0080	libc-2.23.so	buffered_vfprintf
3	0.0060	libc-2.23.so	_IO_file_write@@GLIBC_2.4
3	0.0060	libc-2.23.so	new_do_write

3	0.0060	pi.O0.g	calculate
2	0.0040	libc-2.23.so	_IO_file_overflow@@GLIBC_2.4
2	0.0040	libc-2.23.so	__overflow
2	0.0040	libc-2.23.so	putchar
2	0.0040	pi.O0.g	MULTIPLY
2	0.0040	pi.O0.g	SET
2	0.0040	pi.O0.g	epilog
1	0.0020	libc-2.23.so	_IO_default_xsputn
1	0.0020	libc-2.23.so	_IO_file_xsputn@@GLIBC_2.4
1	0.0020	libc-2.23.so	fprintf
1	0.0020	libc-2.23.so	strchrnul
1	0.0020	pi.O0.g	progress

7)

**a)**

Las principales diferencias están en la columna samples y en las funciones de las que se han tomado muestras (samples). Con `opreport` y `-O3` vemos que las funciones `DIVIDE`, `SUBTRACT`, etc, no son mostradas. Esto es porque no hay llamadas hacia estas funciones, puesto que al haber hecho `inline` se ha copiado el código de las funciones donde se llamaban anteriormente. Por ello, con `opannotate` vemos que para `-O0` si vemos el recuento de muestras justo donde empieza cada una de las funciones nombradas anteriormente, pero no para `-O3`. En cambio si hay recuento del código de cada una de ellas. Porque el código igualmente se ejecuta. Por último, hay diferencias de recuentos en los "for" por una optimización del `-O3` sobre los bucles. Esto provoca una disminución de recuento sobre ellos.

```

%%%%%%%%%%%%%%
%%%%%%%%%%%%%%

```

```
$ opannotate --source pi.O0.g
```

Using `/home/ubuntu/lab2_session/1.2/opprofile_data/samples/` for session-dir

```

/*
 * Command line: opannotate --source pi.O0.g
 *
 * Interpretation of command line:
 * Output annotated source file with samples
 * Output all files
 *
 * CPU: ARM Cortex-A9, speed 999 MHz (estimated)
 * Counted CPU_CYCLES events (CPU cycle) with a unit mask of 0x00 (No unit mask)
count 100000
*/
/*
 * Total samples for file : "/home/ubuntu/lab2_session/1.2/pi.c"
 *
 * 239215 61.7085
 */

```

```

#include <memory.h>
#include <stdio.h>
#include <stdlib.h>

```

```

:
: int N, N4;
: signed char a[25480], b[25480], c[25480];
:
: void DIVIDE( signed char *x, int n )
8 0.0021 :{ /* DIVIDE total: 94519 24.3824 */
:   int j, k;
:   unsigned q, r, u;
:   long v;
:
:   r = 0;
23800 6.1395 :   for( k = 0; k <= N4; k++ )
:   {
21718 5.6024 :       u = r * 10 + x[k];
20615 5.3179 :       q = u/n;
14237 3.6726 :       r = u - q * n;
14137 3.6468 :       x[k] = q;
:   }
4 0.0010 :}
:
: void LONGDIV( signed char *x, int n )
2 5.2e-04 :{ /* LONGDIV total: 34861 8.9928 */
:   int j, k;
:   unsigned q, r, u;
:   long v;
:
:   if( n < 6553 )
:   {
:       r = 0;
2663 0.6870 :       for( k = 0; k <= N4; k++ )
:       {
2640 0.6810 :           u = r * 10 + x[k];
2262 0.5835 :           q = u / n;
1499 0.3867 :           r = u - q * n;
1471 0.3795 :           x[k] = q;
:       }
:   }
:   else
:   {
:       r = 0;
6151 1.5867 :       for( k = 0; k <= N4; k++ )
:       {
2581 0.6658 :           if( r < 6553 )
:           {
4328 1.1165 :               u = r * 10 + x[k];
1846 0.4762 :               q = u / n;
1500 0.3869 :               r = u - q * n;
:           }
:           else
:           {
3320 0.8564 :               v = (long) r * 10 + x[k];
1037 0.2675 :               q = v / n;

```

```

940 0.2425 :      r = v - q * n;
      :      }
2619 0.6756 :      x[k] = q;
      :      }
      :      }
2 5.2e-04 :}
      :
      :void MULTIPLY( signed char *x, int n )
      :{
      :      /* MULTIPLY total:  10  0.0026 */
      :      int j, k;
      :      unsigned q, r, u;
      :      long v;
      :      r = 0;
      :      for( k = N4; k >= 0; k-- )
      :      {
1 2.6e-04 :      q = n * x[k] + r;
2 5.2e-04 :      r = q / 10;
7 0.0018 :      x[k] = q - r * 10;
      :      }
      :}
      :
      :void SET( signed char *x, int n )
5 0.0013 :{
      :      /* SET total:  7  0.0018 */
      :      memset( x, 0, N4 + 1 );
1 2.6e-04 :      x[0] = n;
1 2.6e-04 :}
      :
      :
      :void SUBTRACT( signed char *x, signed char *y, signed char *z )
3 7.7e-04 :{
      :      /* SUBTRACT total: 109775 28.3179 */
      :      int j, k;
      :      unsigned q, r, u;
      :      long v;
24639 6.3559 :      for( k = N4; k >= 1; k-- )
      :      {
46029 11.8738 :      if( (x[k] = y[k] - z[k]) < 0 )
      :      {
25189 6.4978 :      x[k] += 10;
13903 3.5865 :      z[k-1]++;
      :      }
      :      }
8 0.0021 :      if( (x[k] = y[k] - z[k]) < 0 )
      :      {
      :      x[k] += 10;
      :      }
4 0.0010 :}
      :
      :
      :void calculate( void );
      :void progress( void );
      :void epilog( void );
      :

```

```

:
: int main( int argc, char *argv[] )
:{
:   N = 10000;
:
:   if( argc > 1 )
:       N = atoi(argv[1]);
:
:   setbuf(stdout, NULL);
:
:   calculate();
:
:   epilog();
:
:   return 0;
:}
:
: void calculate( void )
:{ /* calculate total:   22 0.0057 */
:   int j;
:
:   N4 = N + 4;
:
:   SET( a, 0 );
:   SET( b, 0 );
:
2 5.2e-04 :   for( j = 2 * N4 + 1; j >= 3; j -= 2 )
:   {
2 5.2e-04 :       SET( c, 1 );
3 7.7e-04 :       LONGDIV( c, j );
:
2 5.2e-04 :       SUBTRACT( a, c, a );
3 7.7e-04 :       DIVIDE( a, 25 );
:
2 5.2e-04 :       SUBTRACT( b, c, b );
4 0.0010 :       DIVIDE( b, 239 );
1 2.6e-04 :       DIVIDE( b, 239 );
:
3 7.7e-04 :       progress();
:   }
:
:   SET( c, 1 );
:
:   SUBTRACT( a, c, a );
:   DIVIDE( a, 5 );
:
:   SUBTRACT( b, c, b );
:   DIVIDE( b, 239 );
:
:   MULTIPLY( a, 4 );
:   SUBTRACT( a, a, b );
:   MULTIPLY( a, 4 );

```



```

:
: progress();
:}
:
:/*
:
: N = 10000
: A = 0
: B = 0
: J = 2 * (N + 4) + 1
: FOR J = J TO 3 STEP -2
:   A = (1 / J - A) / 5 ^ 2
:   B = (1 / J - B) / 239 ^ 2
: NEXT J
: A = (1 - A) / 5
: B = (1 - B) / 239
: PI = (A * 4 - B) * 4
:
:*/
:
: void progress( void )
3 7.7e-04 :{ /* progress total:    5 0.0013 */
:   printf(".");
2 5.2e-04 :}
:
: void epilog( void )
: { /* epilog total:   16 0.0041 */
:   int j;
:
:   {
:     fprintf( stdout, " \n3.");
4 0.0010 :     for( j = 1; j <= N; j++ )
:       {
5 0.0013 :         fprintf( stdout, "%d", a[j]);
7 0.0018 :         if( j % 5 == 0 )
:           if( j % 50 == 0 )
:             if( j % 250 == 0 )
:               fprintf( stdout, "  <%d>\n\n ", j );
:             else
:               fprintf( stdout, "\n  " );
:             else
:               fprintf( stdout, " " );
:           }
:         }
:       }
:     }
:   }
:
:
opannotate (warning): unable to open for reading:
/build/glibc-XzEjT5/glibc-2.23/string/../../sysdeps/arm/memset.S
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/string/../../sysdeps/arm/memset.S"
*
* 199 0.0513

```

\*/

/\* memset total: 199 0.0513 \*/

opannotate (warning): unable to open for reading:  
/build/glibc-XzEjT5/glibc-2.23/libio/fileops.c

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/libio/fileops.c"

\*

\* 115 0.0297

\*/

/\* \_IO\_do\_write@@GLIBC\_2.4 total: 10 0.0026 \*/

/\* new\_do\_write total: 34 0.0088 \*/

/\* \_IO\_file\_overflow@@GLIBC\_2.4 total: 16 0.0041 \*/

/\* \_IO\_file\_write@@GLIBC\_2.4 total: 39 0.0101 \*/

/\* \_IO\_file\_xsputn@@GLIBC\_2.4 total: 16 0.0041 \*/

opannotate (warning): unable to open for reading: /build/glibc-XzEjT5/glibc-2.23/stdio-common/vfprintf.c

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/stdio-common/vfprintf.c"

\*

\* 79 0.0204

\*/

/\* fprintf total: 67 0.0173 \*/

/\* buffered\_vfprintf total: 14 0.0036 \*/

opannotate (warning): unable to open for reading:  
/build/glibc-XzEjT5/glibc-2.23/libio/putchar.c

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/libio/putchar.c"

\*

\* 24 0.0062

\*/

/\* putchar total: 31 0.0080 \*/

opannotate (warning): unable to open for reading:  
/build/glibc-XzEjT5/glibc-2.23/io../sysdeps/unix/syscall-template.S

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/io../sysdeps/unix/syscall-template.S"

\*

\* 23 0.0059

\*/

/\* open total: 1 2.6e-04 \*/

/\* write total: 22 0.0057 \*/

opannotate (warning): unable to open for reading:

/build/glibc-XzEjT5/glibc-2.23/libio/genops.c

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/libio/genops.c"

\*

\* 15 0.0039

\*/

/\* \_\_overflow total: 6 0.0015 \*/

/\* \_IO\_default\_xsputn total: 8 0.0021 \*/

/\* \_IO\_flush\_all\_lockp total: 1 2.6e-04 \*/

opannotate (warning): unable to open for reading:

/build/glibc-XzEjT5/glibc-2.23/string/strchrnul.c

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/string/strchrnul.c"

\*

\* 11 0.0028

\*/

/\* strchrnul total: 11 0.0028 \*/

opannotate (warning): unable to open for reading:

/build/glibc-XzEjT5/glibc-2.23/libio/libioP.h

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/libio/libioP.h"

\*

\* 9 0.0023

\*/

opannotate (warning): unable to open for reading: /build/glibc-XzEjT5/glibc-2.23/stdio-common/\_itoa.c

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/stdio-common/\_itoa.c"

\*

\* 6 0.0015

\*/

/\* \_itoa\_word total: 6 0.0015 \*/

opannotate (warning): unable to open for reading:

/build/glibc-XzEjT5/glibc-2.23/libio/fputc.c

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/libio/fputc.c"

\*

\* 3 7.7e-04

\*/

/\* fputc total: 4 0.0010 \*/

opannotate (warning): unable to open for reading: /build/glibc-XzEjT5/glibc-2.23/stdio-common/printf-parse.h

```
/*
 * Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/stdio-common/printf-parse.h"
 *
 *    2 5.2e-04
 */
```

opannotate (warning): unable to open for reading: /build/glibc-XzEjT5/glibc-2.23/elf/do-rel.h

```
/*
 * Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/elf/do-rel.h"
 *
 *    1 2.6e-04
 */
```

opannotate (warning): unable to open for reading: /build/glibc-XzEjT5/glibc-2.23/elf/dl-lookup.c

```
/*
 * Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/elf/dl-lookup.c"
 *
 *    1 2.6e-04
 */
```

/\* check\_match total: 1 2.6e-04 \*/

%%  
%%

```
$ gcc -g -O3 pi.c -o pi.O3.g
$ perf --event=CPU_CYCLES:100000 ./pi.O3.g
$ opreport -l
```

CPU: ARM Cortex-A9, speed 999 MHz (estimated)

Counted CPU\_CYCLES events (CPU cycle) with a unit mask of 0x00 (No unit mask) count 100000

samples	%	image name	symbol name
90344	69.4500	pi.O3.g	calculate
20858	16.0341	pi.O3.g	__aeabi_uidiv
13123	10.0880	pi.O3.g	.divsi3_skip_div0_test
4551	3.4985	no-vmlinux	/no-vmlinux
481	0.3698	pi.O3.g	__divsi3
367	0.2821	libc-2.23.so	memset
75	0.0577	libc-2.23.so	vfprintf
60	0.0461	libc-2.23.so	putchar
37	0.0284	libc-2.23.so	_IO_file_write@@GLIBC_2.4
28	0.0215	libc-2.23.so	write
26	0.0200	libc-2.23.so	new_do_write
23	0.0177	libc-2.23.so	_IO_file_overflow@@GLIBC_2.4

14	0.0108	libc-2.23.so	__memset_chk
12	0.0092	libc-2.23.so	__fprintf_chk
12	0.0092	libc-2.23.so	buffered_vfprintf
12	0.0092	libc-2.23.so	strchrnul
10	0.0077	libc-2.23.so	_IO_default_xsputn
10	0.0077	libc-2.23.so	_itoa_word
9	0.0069	libc-2.23.so	_IO_do_write@@GLIBC_2.4
9	0.0069	libc-2.23.so	_IO_file_xsputn@@GLIBC_2.4
8	0.0061	libc-2.23.so	__overflow
6	0.0046	libc-2.23.so	___GI_memset_from_thumb
4	0.0031	libc-2.23.so	fputc
4	0.0031	pi.O3.g	epilog
1	7.7e-04	ld-2.23.so	_dl_relocate_object
1	7.7e-04	libc-2.23.so	fwrite

%%%

**\$ opannotate --source pi.O3.g**

Using /home/ubuntu/lab2\_session/1.2/oprofile\_data/samples/ for session-dir

/\*

\* Command line: opannotate --source pi.O3.g

\*

\* Interpretation of command line:

\* Output annotated source file with samples

\* Output all files

\*

\* CPU: ARM Cortex-A9, speed 999 MHz (estimated)

\* Counted CPU\_CYCLES events (CPU cycle) with a unit mask of 0x00 (No unit mask)

count 100000

\*/

/\*

\* Total samples for file : "/home/ubuntu/lab2\_session/1.2/pi.c"

\*

\* 90339 69.4461

\*/

```
#include <memory.h>
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
:
```

```
:int N, N4;
```

```
:signed char a[25480], b[25480], c[25480];
```

```
:
```

```
:void DIVIDE( signed char *x, int n )
```

```
:{
```

```
: int j, k;
```

```
: unsigned q, r, u;
```

```
: long v;
```

```
:
```

```
: r = 0;
```

```

9401 7.2268 :   for( k = 0; k <= N4; k++ )
                :   {
6593 5.0682 :       u = r * 10 + x[k];
6196 4.7630 :       q = u/n;
6221 4.7823 :       r = u - q * n;
18625 14.3176 :       x[k] = q;
                :   }
                : }
                :
                : void LONGDIV( signed char *x, int n )
                : {
                :   int j, k;
                :   unsigned q, r, u;
                :   long v;
                :
2   0.0015 :   if( n < 6553 )
                :   {
                :       r = 0;
805 0.6188 :       for( k = 0; k <= N4; k++ )
                :       {
858 0.6596 :           u = r * 10 + x[k];
1021 0.7849 :           q = u / n;
45  0.0346 :           r = u - q * n;
108 0.0830 :           x[k] = q;
                :       }
                :   }
                :   else
                :   {
                :       r = 0;
1267 0.9740 :       for( k = 0; k <= N4; k++ )
                :       {
2097 1.6120 :           if( r < 6553 )
                :           {
2741 2.1071 :               u = r * 10 + x[k];
888 0.6826 :               q = u / n;
165 0.1268 :               r = u - q * n;
                :           }
                :           else
                :           {
2   0.0015 :               v = (long) r * 10 + x[k];
1185 0.9109 :               q = v / n;
                :               r = v - q * n;
                :           }
204 0.1568 :               x[k] = q;
                :       }
                :   }
                : }
                :
                : void MULTIPLY( signed char *x, int n )
                : {
                :   int j, k;
                :   unsigned q, r, u;

```

```

: long v;
: r = 0;
2 0.0015 : for( k = N4; k >= 0; k-- )
: {
1 7.7e-04 : q = n * x[k] + r;
: r = q / 10;
1 7.7e-04 : x[k] = q - r * 10;
: }
:}
:
: void SET( signed char *x, int n )
:{
: memset( x, 0, N4 + 1 );
1 7.7e-04 : x[0] = n;
:}
:
:
: void SUBTRACT( signed char *x, signed char *y, signed char *z )
:{
: int j, k;
: unsigned q, r, u;
: long v;
4409 3.3893 : for( k = N4; k >= 1; k-- )
: {
11613 8.9272 : if( (x[k] = y[k] - z[k]) < 0 )
: {
2630 2.0218 : x[k] += 10;
13241 10.1787 : z[k-1]++;
: }
: }
: if( (x[k] = y[k] - z[k]) < 0 )
: {
4 0.0031 : x[k] += 10;
: }
:}
:
:
:
: void calculate( void );
: void progress( void );
: void epilog( void );
:
:
: int main( int argc, char *argv[] )
:{
: N = 10000;
:
: if( argc > 1 )
: N = atoi(argv[1]);
:
: setbuf(stdout, NULL);
:
: calculate();

```

```

:
:   epilog();
:
:   return 0;
:}
:
:
: void calculate( void )
: {
:   int j;
:
:   N4 = N + 4; /* calculate total: 90344 69.4500 */
:
:   SET( a, 0 );
:   SET( b, 0 );
:
9 0.0069 :   for( j = 2 * N4 + 1; j >= 3; j -= 2 )
:   {
:     SET( c, 1 );
:     LONGDIV( c, j );
:
:     SUBTRACT( a, c, a );
:     DIVIDE( a, 25 );
:
:     SUBTRACT( b, c, b );
:     DIVIDE( b, 239 );
:     DIVIDE( b, 239 );
:
:     progress();
:   }
:
:   SET( c, 1 );
:
:   SUBTRACT( a, c, a );
:   DIVIDE( a, 5 );
:
:   SUBTRACT( b, c, b );
:   DIVIDE( b, 239 );
:
:   MULTIPLY( a, 4 );
:   SUBTRACT( a, a, b );
:   MULTIPLY( a, 4 );
:
:   progress();
: }
:
: /*
:
: N = 10000
: A = 0
: B = 0
: J = 2 * (N + 4) + 1
: FOR J = J TO 3 STEP -2

```





```

/* _IO_do_write@@GLIBC_2.4 total:   9 0.0069 */
/* new_do_write total:   26 0.0200 */
/* _IO_file_overflow@@GLIBC_2.4 total:   23 0.0177 */
/* _IO_file_write@@GLIBC_2.4 total:   37 0.0284 */
/* _IO_file_xsputn@@GLIBC_2.4 total:   9 0.0069 */
opannotate (warning): unable to open for reading: /build/glibc-XzEjT5/glibc-2.23/stdio-
common/vfprintf.c
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/stdio-common/vfprintf.c"
*
*   86 0.0661
*/

/* fprintf total:   75 0.0577 */
/* buffered_vfprintf total:   12 0.0092 */
opannotate (warning): unable to open for reading:
/build/glibc-XzEjT5/glibc-2.23/libio/putchar.c
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/libio/putchar.c"
*
*   51 0.0392
*/

/* putchar total:   60 0.0461 */
opannotate (warning): unable to open for reading:
/build/glibc-XzEjT5/glibc-2.23/io/./sysdeps/unix/syscall-template.S
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/io/./sysdeps/unix/syscall-
template.S"
*
*   28 0.0215
*/

/* write total:   28 0.0215 */
opannotate (warning): unable to open for reading: /
build/glibc-XzEjT5/glibc-2.23/libio/genops.c
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/libio/genops.c"
*
*   18 0.0138
*/

/* __overflow total:   8 0.0061 */
/* _IO_default_xsputn total:   10 0.0077 */
opannotate (warning): unable to open for reading:
/build/glibc-XzEjT5/glibc-2.23/debug/memset_chk.c

```

```

/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/debug/memset_chk.c"
*
* 14 0.0108
*/

/* __memset_chk total: 14 0.0108 */
opannotate (warning): unable to open for reading:
/build/glibc-XzEjT5/glibc-2.23/string/strchrnul.c
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/string/strchrnul.c"
*
* 12 0.0092
*/

/* strchrnul total: 12 0.0092 */
opannotate (warning): unable to open for reading: /build/glibc-XzEjT5/glibc-2.23/stdio-
common/_itoa.c
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/stdio-common/_itoa.c"
*
* 10 0.0077
*/

/* _itoa_word total: 10 0.0077 */
opannotate (warning): unable to open for reading: /
/build/glibc-XzEjT5/glibc-2.23/libio/libioP.h
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/libio/libioP.h"
*
* 10 0.0077
*/

/*
* Total samples for file : "/usr/include/arm-linux-gnueabi/hf/bits/string3.h"
*
* 9 0.0069
*/

```

```

/* Copyright (C) 2004-2016 Free Software Foundation, Inc.
: This file is part of the GNU C Library.
:
: The GNU C Library is free software; you can redistribute it and/or
: modify it under the terms of the GNU Lesser General Public
: License as published by the Free Software Foundation; either
: version 2.1 of the License, or (at your option) any later version.
:

```

```
: The GNU C Library is distributed in the hope that it will be useful,  
: but WITHOUT ANY WARRANTY; without even the implied warranty of  
: MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See
```

the GNU

```
: Lesser General Public License for more details.
```

```
:
```

```
: You should have received a copy of the GNU Lesser General Public
```

```
: License along with the GNU C Library; if not, see
```

```
: <http://www.gnu.org/licenses/>. */
```

```
:
```

```
:#ifndef _STRING_H
```

```
:# error "Never use <bits/string3.h> directly; include <string.h> instead."
```

```
:#endif
```

```
:
```

```
:#if !__GNUC_PREREQ (5,0)
```

```
:__warndecl (__warn_memset_zero_len,
```

```
: "memset used with constant zero length parameter; this could be due to  
transposed parameters");
```

```
:#endif
```

```
:
```

```
:#ifndef __cplusplus
```

```
:/ * XXX This is temporarily. We should not redefine any of the symbols
```

```
: and instead integrate the error checking into the original
```

```
: definitions. */
```

```
:# undef memcpy
```

```
:# undef memmove
```

```
:# undef memset
```

```
:# undef strcat
```

```
:# undef strcpy
```

```
:# undef strncat
```

```
:# undef strncpy
```

```
:# ifdef __USE_GNU
```

```
:# undef memcpy
```

```
:# undef stpcpy
```

```
:# endif
```

```
:# ifdef __USE_MISC
```

```
:# undef bcopy
```

```
:# undef bzero
```

```
:# endif
```

```
:#endif
```

```
:
```

```
:
```

```
:__fortify_function void *
```

```
:__NTH (memcpy (void *__restrict __dest, const void *__restrict __src,
```

```
: size_t __len))
```

```
:{
```

```
: return __builtin___memcpy_chk (__dest, __src, __len, __bos0 (__dest));
```

```
;} 
```

```
:
```

```
:__fortify_function void *
```

```
:__NTH (memmove (void *__dest, const void *__src, size_t __len))
```

```
:{
```

```

: return __builtin__memmove_chk (__dest, __src, __len, __bos0 (__dest));
:}
:
:
: #ifdef __USE_GNU
: __fortify_function void *
: __NTH (memcpy (void *__restrict __dest, const void *__restrict __src,
:             size_t __len))
: {
: return __builtin__memcpy_chk (__dest, __src, __len, __bos0 (__dest));
: }
: #endif
:
:
: /* The first two tests here help to catch a somewhat common problem
:  * where the second and third parameter are transposed. This is
:  * especially problematic if the intended fill value is zero. In this
:  * case no work is done at all. We detect these problems by referring
:  * non-existing functions. */
: __fortify_function void *
: __NTH (memset (void *__dest, int __ch, size_t __len))
: {
: /* GCC-5.0 and newer implements these checks in the compiler, so we don't
:  * need them here. */
: #if !__GNUC_PREREQ (5,0)
: if (__builtin_constant_p (__len) && __len == 0
:     && (!__builtin_constant_p (__ch) || __ch != 0))
: {
:   __warn_memset_zero_len ();
:   return __dest;
: }
: #endif
9 0.0069: return __builtin__memset_chk (__dest, __ch, __len, __bos0 (__dest));
:}
:
:
: #ifdef __USE_MISC
: __fortify_function void
: __NTH (bcopy (const void *__src, void *__dest, size_t __len))
: {
: (void) __builtin__memmove_chk (__dest, __src, __len, __bos0 (__dest));
: }
:
:
: __fortify_function void
: __NTH (bzero (void *__dest, size_t __len))
: {
: (void) __builtin__memset_chk (__dest, '\0', __len, __bos0 (__dest));
: }
: #endif
:
:
: __fortify_function char *
: __NTH (strcpy (char *__restrict __dest, const char *__restrict __src))
: {
: return __builtin__strcpy_chk (__dest, __src, __bos (__dest));

```

```

:}
:
:
:ifndef __USE_GNU
:__fortify_function char *
:__NTH (strcpy (char *__restrict __dest, const char *__restrict __src))
:{
: return __builtin___strcpy_chk (__dest, __src, __bos (__dest));
:}
:endif
:
:
:
:__fortify_function char *
:__NTH (strncpy (char *__restrict __dest, const char *__restrict __src,
:               size_t __len))
:{
: return __builtin___strncpy_chk (__dest, __src, __len, __bos (__dest));
:}
:
:
:// XXX We have no corresponding builtin yet.
:extern char *__stpncpy_chk (char *__dest, const char *__src, size_t __n,
:                             size_t __destlen) __THROW;
:extern char *__REDIRECT_NTH (__stpncpy_alias, (char *__dest, const char
*__src,
:
:                             size_t __n), stpncpy);
:
:
:__fortify_function char *
:__NTH (stpncpy (char *__dest, const char *__src, size_t __n))
:{
: if (__bos (__dest) != (size_t) -1
:     && (!__builtin_constant_p (__n) || __n > __bos (__dest)))
:   return __stpncpy_chk (__dest, __src, __n, __bos (__dest));
: return __stpncpy_alias (__dest, __src, __n);
:}
:
:
:
:
:__fortify_function char *
:__NTH (strcat (char *__restrict __dest, const char *__restrict __src))
:{
: return __builtin___strcat_chk (__dest, __src, __bos (__dest));
:}
:
:
:
:__fortify_function char *
:__NTH (strncat (char *__restrict __dest, const char *__restrict __src,
:               size_t __len))
:{
: return __builtin___strncat_chk (__dest, __src, __len, __bos (__dest));
:}

```

opannotate (warning): unable to open for reading:  
/build/glibc-XzEjT5/glibc-2.23/debug/fprintf\_chk.c

/\*

\* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/debug/fprintf\_chk.c"

```
*
* 9 0.0069
*/
```

```
/* __fprintf_chk total: 12 0.0092 */
opannotate (warning): unable to open for reading:
/build/glibc-XzEjT5/glibc-2.23/libio/fputc.c
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/libio/fputc.c"
*
* 4 0.0031
*/
```

```
/* fputc total: 4 0.0031 */
opannotate (warning): unable to open for reading:
/build/glibc-XzEjT5/glibc-2.23/debug/../libio/libioP.h
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/debug/../libio/libioP.h"
*
* 3 0.0023
*/
```

```
opannotate (warning): unable to open for reading: /build/glibc-XzEjT5/glibc-2.23/stdio-
common/printf-parse.h
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/stdio-common/printf-parse.h"
*
* 1 7.7e-04
*/
```

```
opannotate (warning): unable to open for reading:
/build/glibc-XzEjT5/glibc-2.23/elf/../sysdeps/arm/dl-machine.h
/*
* Total samples for file : "/build/glibc-XzEjT5/glibc-2.23/elf/../sysdeps/arm/dl-machine.h"
*
* 1 7.7e-04
*/
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

8)

a)

Observamos que, dependiendo de la frecuencia que le pongamos, se ejecutan más samples, aun así el cuello de botella que genera cada rutina más o menos es el mismo. PD: El máximo que nos dejaba poner como frecuencia eran 32200.

```
$ sudo perf record -e cycles -F 32200 ./pi.O0.g 1000
```

```
$ sudo perf report --stdio -n
```

```
# To display the perf.data header info, please use --header/--he
#
#
# Total Lost Samples: 0
#
# Samples: 25K of event 'cycles'
# Event count (approx.): 514641219
#
# Overhead      Samples Command Shared Object      Symbol
# .....
#
33.81%      8499 pi.O0.g pi.O0.g      [.] __udivs
25.59%      6429 pi.O0.g pi.O0.g      [.] SUBTRAC
22.01%      5532 pi.O0.g pi.O0.g      [.] DIVIDE
8.11%       2041 pi.O0.g pi.O0.g      [.] LONGDIV
2.29%        580 pi.O0.g [kernel.kallsyms] [k] queue_w
0.96%        242 pi.O0.g [kernel.kallsyms] [k] __do_so
0.74%        186 pi.O0.g [kernel.kallsyms] [k] v7_dma_
0.33%         84 pi.O0.g [kernel.kallsyms] [k] l2c210_
0.28%         71 pi.O0.g [kernel.kallsyms] [k] tcp_ack
0.25%         64 pi.O0.g [kernel.kallsyms] [k] _raw_sp
0.25%         63 pi.O0.g libc-2.23.so [.] vfprint
0.25%         62 pi.O0.g [kernel.kallsyms] [k] n_tty_w
0.19%         48 pi.O0.g [kernel.kallsyms] [k] tty_wri
0.15%         38 pi.O0.g [kernel.kallsyms] [k] ip_rcv
0.13%         32 pi.O0.g [kernel.kallsyms] [k] tcp_v4_
0.12%         30 pi.O0.g [kernel.kallsyms] [k] __netde
0.11%         29 pi.O0.g [kernel.kallsyms] [k] vector_
0.11%         28 pi.O0.g [kernel.kallsyms] [k] __netif
0.09%         23 pi.O0.g [kernel.kallsyms] [k] dma_cac
0.09%         22 pi.O0.g [kernel.kallsyms] [k] gem_rx_
0.09%         22 pi.O0.g [kernel.kallsyms] [k] vfs_wri
0.08%         20 pi.O0.g libc-2.23.so [.] __GI__
0.08%         20 pi.O0.g libc-2.23.so [.] new_do_
0.08%         19 pi.O0.g [kernel.kallsyms] [k] skb_rel
0.07%         19 pi.O0.g [kernel.kallsyms] [k] tty_par
0.07%         17 pi.O0.g libc-2.23.so [.] putchar
0.07%         17 pi.O0.g pi.O0.g      [.] calcula
0.07%         17 pi.O0.g [kernel.kallsyms] [k] __l2c21
0.07%         17 pi.O0.g [kernel.kallsyms] [k] macb_po
0.07%         17 pi.O0.g [kernel.kallsyms] [k] __do_di
0.07%         17 pi.O0.g libc-2.23.so [.] buffere
0.06%         16 pi.O0.g [kernel.kallsyms] [k] eth_typ
0.06%         16 pi.O0.g [kernel.kallsyms] [k] tty_ins
0.06%         16 pi.O0.g [kernel.kallsyms] [k] __fdget
0.06%         16 pi.O0.g libc-2.23.so [.] _IO_fil
0.06%         15 pi.O0.g [kernel.kallsyms] [k] tcp_rcv
0.06%         15 pi.O0.g libc-2.23.so [.] _IO_fil
0.06%         15 pi.O0.g [kernel.kallsyms] [k] mmioset
```



```
0.06%      14 pi.O0.g [kernel.kallsyms] [k] skb_put
0.06%      14 pi.O0.g [kernel.kallsyms] [k] kfree
:
```

```
#
# (Cannot load tips.txt file, please install perf!)
#
```

```
%%%%%%%%%
```

```
$ sudo perf record -e cycles -F 5000 ./pi.O0.g 1000
$ sudo perf report --stdio -n
```

```
# To display the perf.data header info, please use --header/--header-only options.
```

```
#
```

```
#
```

```
# Total Lost Samples: 0
```

```
#
```

```
# Samples: 3K of event 'cycles'
```

```
# Event count (approx.): 434938242
```

```
#
```

```
# Overhead    Samples Command Shared Object    Symbol
```

```
# .....  
#
```

34.17%	1117	pi.O0.g	pi.O0.g	[.]	__udivsi3
26.11%	854	pi.O0.g	pi.O0.g	[.]	SUBTRACT
21.44%	701	pi.O0.g	pi.O0.g	[.]	DIVIDE
7.09%	232	pi.O0.g	pi.O0.g	[.]	LONGDIV
1.70%	65	pi.O0.g	[kernel.kallsyms]	[k]	queue_work_on
1.48%	52	pi.O0.g	[kernel.kallsyms]	[k]	__do_softirq
0.88%	31	pi.O0.g	[kernel.kallsyms]	[k]	v7_dma_inv_range
0.43%	16	pi.O0.g	[kernel.kallsyms]	[k]	n_tty_write
0.31%	11	pi.O0.g	[kernel.kallsyms]	[k]	tcp_ack
0.30%	11	pi.O0.g	[kernel.kallsyms]	[k]	l2c210_inv_range
0.22%	9	pi.O0.g	[kernel.kallsyms]	[k]	_raw_spin_unlock_irqrestore
0.22%	8	pi.O0.g	[kernel.kallsyms]	[k]	tcp_v4_rcv
0.22%	8	pi.O0.g	[kernel.kallsyms]	[k]	ip_rcv
0.21%	9	pi.O0.g	libc-2.23.so	[.]	vfprintf
0.19%	7	pi.O0.g	[kernel.kallsyms]	[k]	dma_cache_maint_page
0.15%	5	pi.O0.g	libc-2.23.so	[.]	memset
0.14%	5	pi.O0.g	[kernel.kallsyms]	[k]	ip_local_deliver
0.14%	5	pi.O0.g	[kernel.kallsyms]	[k]	gem_rx_refill
0.14%	5	pi.O0.g	[kernel.kallsyms]	[k]	tty_write
0.12%	4	pi.O0.g	[kernel.kallsyms]	[k]	__kfree_skb
0.12%	4	pi.O0.g	[kernel.kallsyms]	[k]	__netdev_alloc_skb
0.11%	3	pi.O0.g	[kernel.kallsyms]	[k]	filemap_map_pages
0.11%	4	pi.O0.g	[kernel.kallsyms]	[k]	gem_rx
0.11%	4	pi.O0.g	[kernel.kallsyms]	[k]	__l2c210_cache_sync
0.09%	3	pi.O0.g	[kernel.kallsyms]	[k]	vfs_write
0.09%	3	pi.O0.g	libc-2.23.so	[.]	putchar
0.09%	3	pi.O0.g	[kernel.kallsyms]	[k]	__rcu_read_unlock

0.09%	3	pi.O0.g	[kernel.kallsyms]	[k]	vector_swi
0.09%	3	pi.O0.g	[kernel.kallsyms]	[k]	tcp_v4_early_demux
0.09%	3	pi.O0.g	[kernel.kallsyms]	[k]	__dma_page_dev_to_cpu
0.09%	3	pi.O0.g	libc-2.23.so	[.]	new_do_write
0.09%	3	pi.O0.g	[kernel.kallsyms]	[k]	mod_timer
0.08%	3	pi.O0.g	[kernel.kallsyms]	[k]	__netif_receive_skb_core
0.08%	3	pi.O0.g	[kernel.kallsyms]	[k]	tcp_xmit_recovery
0.08%	3	pi.O0.g	[kernel.kallsyms]	[k]	__fdget_pos
0.08%	3	pi.O0.g	[kernel.kallsyms]	[k]	__memzero
0.08%	3	pi.O0.g	[kernel.kallsyms]	[k]	netif_receive_skb_internal
0.08%	3	pi.O0.g	[kernel.kallsyms]	[k]	__vfs_write
0.07%	3	pi.O0.g	[kernel.kallsyms]	[k]	__tty_buffer_request_room
0.06%	2	pi.O0.g	pi.O0.g	[.]	progress

9)

a)

Vemos como para la misma frecuencia se ejecutan muchos menos samples, por lo que comentamos en ejercicios anteriores. Nos ahorramos las llamadas a las funciones que han le han hecho inline. Por lo que nos ahorramos ese recuento de samples. Tambien por las optimizaciones de los bucles. EL cuello de botella, sin embargo, es mayor en calculate que antes, por el mismo motivo (inline) al ejecutar el código de las funciones, que se les ha hecho inline, dentro de la rutina calculate, aumenta el código a ejecutar y por lo tanto genera más cuello de botella.

```
$ sudo perf record -e cycles -F 32200 ./pi.O3.g 1000
```

```
$ sudo perf report --stdio -n
```

```
# To display the perf.data header info, please use --header/--header-only options.
```

```
#
```

```
#
```

```
# Total Lost Samples: 0
```

```
#
```

```
# Samples: 8K of event 'cycles'
```

```
# Event count (approx.): 166145838
```

```
#
```

```
# Overhead    Samples Command Shared Object    Symbol
```

```
# ..... .....
```

```
#
```

61.88%	5061	pi.O3.g	pi.O3.g	[.]	calculate
23.55%	1925	pi.O3.g	pi.O3.g	[.]	__udivsi3
5.40%	461	pi.O3.g	[kernel.kallsyms]	[k]	queue_work_on
0.72%	62	pi.O3.g	[kernel.kallsyms]	[k]	n_tty_write
0.61%	53	pi.O3.g	[kernel.kallsyms]	[k]	_raw_spin_unlock_irqrestore
0.45%	42	pi.O3.g	libc-2.23.so	[.]	vfprintf
0.38%	33	pi.O3.g	[kernel.kallsyms]	[k]	tty_write
0.33%	27	pi.O3.g	libc-2.23.so	[.]	memset
0.29%	24	pi.O3.g	libc-2.23.so	[.]	__GI___libc_write
0.28%	25	pi.O3.g	libc-2.23.so	[.]	new_do_write
0.22%	18	pi.O3.g	libc-2.23.so	[.]	putchar
0.21%	18	pi.O3.g	[kernel.kallsyms]	[k]	vfs_write
0.21%	18	pi.O3.g	[kernel.kallsyms]	[k]	vector_swi

0.19%	18	pi.O3.g	[kernel.kallsyms]	[k]	v7_dma_inv_range
0.18%	17	pi.O3.g	libc-2.23.so	[.]	_IO_file_overflow@@GLIBC_2.4
0.18%	15	pi.O3.g	[kernel.kallsyms]	[k]	mutex_lock
0.18%	15	pi.O3.g	libc-2.23.so	[.]	_IO_file_write@@GLIBC_2.4
0.16%	15	pi.O3.g	[kernel.kallsyms]	[k]	__do_softirq
0.14%	14	pi.O3.g	[kernel.kallsyms]	[k]	tcp_ack
0.13%	11	pi.O3.g	[kernel.kallsyms]	[k]	process_echoes
0.11%	9	pi.O3.g	[kernel.kallsyms]	[k]	sys_write
0.10%	9	pi.O3.g	[kernel.kallsyms]	[k]	__tty_buffer_request_room
0.10%	10	pi.O3.g	[kernel.kallsyms]	[k]	fsnotify
0.10%	11	pi.O3.g	libc-2.23.so	[.]	buffered_vfprintf
0.07%	7	pi.O3.g	[kernel.kallsyms]	[k]	tty_insert_flip_string_fixed_flag
0.07%	6	pi.O3.g	[kernel.kallsyms]	[k]	v7_flush_icache_all
0.07%	6	pi.O3.g	[kernel.kallsyms]	[k]	rw_verify_area
0.07%	6	pi.O3.g	[kernel.kallsyms]	[k]	__fget_light
0.07%	6	pi.O3.g	[kernel.kallsyms]	[k]	mmiocpy
0.07%	6	pi.O3.g	[kernel.kallsyms]	[k]	add_wait_queue
0.07%	6	pi.O3.g	[kernel.kallsyms]	[k]	tty_paranoia_check
0.07%	6	pi.O3.g	[kernel.kallsyms]	[k]	tty_write_lock
0.07%	7	pi.O3.g	[kernel.kallsyms]	[k]	get_seconds
0.07%	6	pi.O3.g	[kernel.kallsyms]	[k]	tty_ldisc_ref_wait
0.06%	26	pi.O3.g	[kernel.kallsyms]	[k]	_raw_spin_unlock_irq
0.06%	5	pi.O3.g	[kernel.kallsyms]	[k]	pty_write
0.06%	5	pi.O3.g	[kernel.kallsyms]	[k]	mutex_trylock
0.06%	5	pi.O3.g	[kernel.kallsyms]	[k]	mmioset
0.06%	5	pi.O3.g	[kernel.kallsyms]	[k]	ldsem_up_read
0.06%	5	pi.O3.g	[kernel.kallsyms]	[k]	filemap_map_pages

10)

a)

El tiempo mostrado por times en pi\_time es el elapsed time.

```
$ gcc -O0 pi_times.c -o pi_times
```

```
$ ./pi_times
```

Timing amb crida times: user 0.560000 segons, system: 0.000000 segons  
with time:

```
$ /usr/bin/time -o pi_time_10 -f "Elapsed time: %e User: %U System: %S " ./pitime.O0 1000
```

Elapsed time: 0.62 User: 0.59 System: 0.02

```
$ /usr/bin/time -o pi_O0 -f "Elapsed time: %e User: %U System: %S " ./pi.O0 1000
```

Elapsed time: 0.59 User: 0.55 System: 0.03

```
$ perf stat ./pi_times 1000 > out_pitimes_perfstat
```

Timing amb crida times: user 0.540000 segons, system: 0.030000 segons

Performance counter stats for './pi\_times 1000':

```
586.056771 task-clock (msec)    # 0.998 CPUs utilized
      1 context-switches        # 0.002 K/sec
```

0	cpu-migrations	#	0.000 K/sec
36	page-faults	#	0.061 K/sec
390,693,570	cycles	#	0.667 GHz
397,726,420	instructions	#	1.02 insn per cycle
20,442,204	branches	#	34.881 M/sec
4,377,279	branch-misses	#	21.41% of all branches

0.587480439 seconds time elapsed