

Daniel Prieto Remacha

Miguel Zayas Boíza

Práctica 3. Perfilado con código fuente

gcc(~40 min.)

gcc (*GNU C and C++ compiler*)

- Optimización de código basada en información proporcionada por perfiles de ejecución (*Profile-Guided Optimization*, PGO, o *Feedback-Directed Optimization*, FDO)
- Uso:
 1. Compilar con `-fprofile-generate` para añadir instrumentación al programa
 2. Ejecutar el programa con distintas cargas para generar información (archivo `*.gcda`)
 3. Compilar con `-fprofile-use` para activar optimizaciones basadas en la información del perfil de ejecución

Obtén los tiempos de ejecución para cada nivel de optimización al procesar la imagen `img.pgm`

```
usuario@debian:~$time ./edges(0-4) img.pgm out.pgm
usuario@debian:~/Escritorio/laboratorio$ time ./edges0 img.pgm out.pgm
real    0m2.462s
user    0m2.456s
sys     0m0.000s
usuario@debian:~/Escritorio/laboratorio$ time ./edges1 img.pgm out.pgm
real    0m1.398s
user    0m1.388s
sys     0m0.008s
usuario@debian:~/Escritorio/laboratorio$ time ./edges2 img.pgm out.pgm
real    0m1.426s
user    0m1.416s
sys     0m0.012s
usuario@debian:~/Escritorio/laboratorio$ time ./edges3 img.pgm out.pgm
real    0m1.182s
user    0m1.172s
sys     0m0.008s
```

Obtén los tiempos de ejecución compilando con optimización basada en perfil.

```
usuario@debian:~$time ./edges(1-3)pu img.pgm out.pgm
usuario@debian:~/Escritorio/laboratorio$ time ./edges1pu img.pgm out.pgm
real    0m1.329s
user    0m1.316s
sys     0m0.016s
usuario@debian:~/Escritorio/laboratorio$ time ./edges2pu img.pgm out.pgm
real    0m0.834s
user    0m0.820s
sys     0m0.012s
usuario@debian:~/Escritorio/laboratorio$ time ./edges3pu img.pgm out.pgm
real    0m0.574s
user    0m0.568s
sys     0m0.004s
```

Se puede ver que después de activar las optimizaciones basadas en la información del perfil de ejecución se han reducido los tiempos de ejecución. En el caso de edges1 el cambio no se nota tanto, sin embargo, en edges2 y edges3 el tiempo casi se reduce a la mitad.

[gprof\(~20 min.\)](#)

gprof (*GNU profiler*)

- Proporciona el tiempo de ejecución de cada función (por **muestreo**) y el número de veces que se ejecuta y desde dónde (por **detección de eventos**)
 - Solo muestrea en tiempo de procesador (temporizador `ITIMER_PROF`), en modo usuario y el hilo principal
 - No muestrea bibliotecas de enlace dinámico
- Uso:
 1. Compilar con `-pg` para añadir instrumentación al programa
 2. Ejecutar el programa para generar información (archivo `gmon.out`)
 3. Generar informes con `gprof`

Consulta la página de manual de gprof.

DESCRIPTION

"gprof" produces an execution profile of C, Pascal, or Fortran77 programs. The effect of called routines is incorporated in the profile of each caller. The profile data is taken from the call graph profile file (gmon.out default) which is created by programs that are compiled with the **-pg** option of "cc", "pc", and "f77". The **-pg** option also links in versions of the library routines that are compiled for profiling. "Gprof" reads the given object file (the default is "a.out") and establishes the relation between its symbol table and the call graph profile from gmon.out. If more than one profile file is specified, the "gprof" output shows the sum of the profile information in the given profile files.

If you use gcc 2.95.x or 3.0 to compile your binaries, you may need to add the **-fprofile-arcs** to the compile command line in order for the call graphs to be properly stored in gmon.out.

"Gprof" calculates the amount of time spent in each routine. Next, these times are propagated along the edges of the call graph. Cycles are discovered, and calls into a cycle are made to share the time of the cycle.

Several forms of output are available from the analysis.

The flat profile shows how much time your program spent in each function, and how many times that function was called. If you simply want to know which functions burn most of the cycles, it is stated concisely here.

-Opciones destacadas:

- -p: muestra un perfil plano
- -b: suprime la impresión de una descripción de cada campo en el perfil.

Obtén un perfil de ejecución del programa edges.c (sin optimización).

```
usuario@debian:~$gcc -o edges edges.c
```

```
usuario@debian:~$./edges img.pgm out.pgm
```

```
usuario@debian:~$gprof -p -b edges
```

```
usuario@debian:~/Escritorio/laboratorio$ gprof -p -b edges
Flat profile:
```

```
Each sample counts as 0.01 seconds.
```

% time	cumulative seconds	self seconds	calls	self s/call	total s/call	name
82.04	2.17	2.17	2	1.08	1.08	gaussian
18.23	2.65	0.48	1	0.48	0.48	laplacian
0.00	2.65	0.00	1	0.00	2.65	edges
0.00	2.65	0.00	1	0.00	0.00	load_image_file
0.00	2.65	0.00	1	0.00	0.00	save_image_file

Obtén el tiempo de ejecución del programa sin instrumentación y con ella. Calcula la sobrecarga producida por la instrumentación como la diferencia entre los tiempos de ejecución con y sin instrumentación, dividida por el tiempo de ejecución con instrumentación y multiplicada por 100, en porcentaje.

Para calcular el tiempo con instrumentación hemos compilado el programa con la opción **-pg** para añadir la instrumentación.

Tiempo sin instrumentación

```
usuario@debian:~$gcc -o edges edges.c
usuario@debian:~$time ./edges img.pgm out.pgm
usuario@debian:~/Escritorio/laboratorio$ time ./edges img.pgm out.pgm

real    0m2.446s
user    0m2.412s
sys     0m0.016s
```

Tiempo con instrumentación

```
usuario@debian:~$gcc -o edges edges.c
usuario@debian:~$time ./edges img.pgm out.pgm
usuario@debian:~/Escritorio/laboratorio$ time ./edges img.pgm out.pgm

real    0m2.671s
user    0m2.628s
sys     0m0.012s
```

Sobrecarga: $(2.671 - 2.446) / 2.671 * 100 = 8.424\%$

Obtén un perfil de ejecución por líneas de código usando la opción -l. Es necesario haber compilado el programa con la opción -g, que añade información de depuración al ejecutable.

```
usuario@debian:~$gcc -g -o edges edges.c
usuario@debian:~$./edges img.pgm out.pgm
usuario@debian:~$gprof -l edges
usuario@debian:~/Escritorio/laboratorio$ gprof -l edges
Flat profile:

Each sample counts as 0.01 seconds.
%   cumulative   self           self         total
time  seconds    seconds   calls   Ts/call   Ts/call   name
50.55      1.21      1.21           3      0.00      0.00   laplacian (edges.c:157 @ 400f38)
17.13      1.62      0.41           1      0.00      0.00   laplacian (edges.c:155 @ 400f5a)
16.71      2.03      0.40           1      0.00      0.00   _libc_csu_init
 5.22      2.15      0.13           1      0.00      0.00   laplacian (edges.c:153 @ 400eee)
 2.72      2.22      0.07           1      0.00      0.00   laplacian (edges.c:151 @ 400f10)
 2.09      2.27      0.05           1      0.00      0.00   main (edges.c:199 @ 401170)
 0.84      2.29      0.02           1      0.00      0.00   fini
 0.84      2.31      0.02           1      0.00      0.00   laplacian (edges.c:152 @ 400ed5)
 0.84      2.33      0.02           1      0.00      0.00   laplacian (edges.c:164 @ 400f8a)
 0.63      2.34      0.02           1      0.00      0.00   laplacian (edges.c:155 @ 400f1f)
 0.63      2.36      0.02           1      0.00      0.00   laplacian (edges.c:156 @ 400f28)
 0.42      2.37      0.01           1      0.00      0.00   gaussian (edges.c:101 @ 400b98)
 0.42      2.38      0.01           1      0.00      0.00   gaussian (edges.c:121 @ 400ce4)
 0.42      2.39      0.01           1      0.00      0.00   laplacian (edges.c:163 @ 400f81)
 0.42      2.40      0.01           1      0.00      0.00   laplacian (edges.c:165 @ 400f96)
 0.42      2.41      0.01           1      0.00      0.00   main (edges.c:202 @ 401193)
 0.00      2.41      0.00           3      0.00      0.00   gaussian (edges.c:100 @ 400b76)
 0.00      2.41      0.00           1      0.00      0.00   laplacian (edges.c:140 @ 400dfb)
 0.00      2.41      0.00           1      0.00      0.00   load_image_file (edges.c:18 @ 40081c)
```

Call graph (explanation follows)

granularity: each sample hit covers 2 byte(s) for 0.42% of 2.41 seconds

index	% time	self	children	called	name
[3]	16.7	0.40	0.00		<spontaneous> __libc_csu_init [3]

				1	_fini [9]
[9]	0.8	0.02	0.00	0+1	_fini [9]
		0.00	0.00	3/3	gaussian (edges.c:100 @ 400b76) [17]
		0.00	0.00	1/1	laplacian (edges.c:140 @ 400dfb) [18]
		0.00	0.00	1/1	load_image_file (edges.c:18 @ 40081c) [19]
				1	_fini [9]

		0.00	0.00	3/3	_fini [9]
[17]	0.0	0.00	0.00	3	gaussian (edges.c:100 @ 400b76) [17]

		0.00	0.00	1/1	_fini [9]
[18]	0.0	0.00	0.00	1	laplacian (edges.c:140 @ 400dfb) [18]

		0.00	0.00	1/1	_fini [9]
[19]	0.0	0.00	0.00	1	load_image_file (edges.c:18 @ 40081c) [19]

google-pprof(~30 min.)

man google-pprof:

DESCRIPCIÓN: Imprime el perfil de CPU o montón especificado

OPCIONES:

--cum	Ordenar por datos acumulativos
--base=<base>	Restar <base> de <perfil> antes de mostrar
Granularidad de informes:	
--addresses	Informe a nivel de dirección
--lines	Informe a nivel de línea de origen
--functions	Informe a nivel de función [predeterminado]
--files	Informe a nivel de archivo de origen
Tipo de salida:	
--text	Generar informe de texto [predeterminado]
--pdf	Generar PDF a la salida estándar
Opciones de perfil de montón:	
--inuse_space	Mostrar (mega)bytes en uso [predeterminado]
--inuse_objects	Mostrar objetos en uso
--alloc_space	Mostrar (mega) bytes asignados
--alloc_objects	Mostrar objetos asignados
--show_bytes	Mostrar espacio en bytes
--drop_negative	Ignorar diferencias negativas

Obtén un perfil de ejecución en formato de texto del programa edges.c (en versiones recientes de gcc es necesario usar la opción -Wl,--no-as-needed para forzar el enlace).

```
usuario@debian:~/Descargas/lab$ gcc -o edges edges.c -Wl,--no-as-needed -lprofiler
usuario@debian:~/Descargas/lab$ CUPROFILE=/tmp/edges.prof ./edges img.pgm out.pgm
usuario@debian:~/Descargas/lab$ google-pprof --text edges /tmp/edges.prof
```

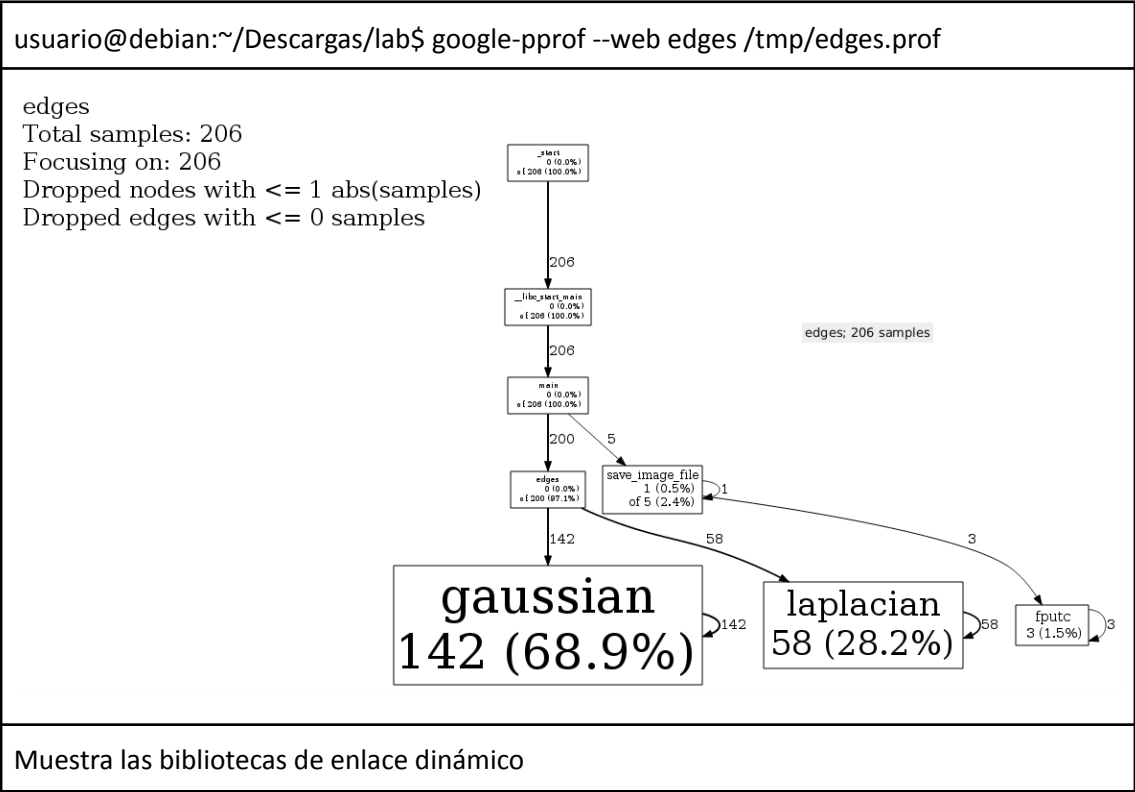
```
Terminal - usuario@debian: ~/Descargas/lab
Archivo Editar Ver Terminal Ir Ayuda
usuario@debian:~/Descargas/lab$ ls
cpu.mem.c  edges1  edges2  edges3pu  edges.gcd  matrix2.c
edges      edges1pg  edges2pu  edges4pu  img.pgm    out.pgm
edges0     edges1pu  edges3    edges.c    matrix1.c
usuario@debian:~/Descargas/lab$ gcc -o edges edges.c
usuario@debian:~/Descargas/lab$ edges img.pgm out.pgm
bash: edges: no se encontró la orden
usuario@debian:~/Descargas/lab$ ./edges img.pgm out.pgm
usuario@debian:~/Descargas/lab$ gprof -p -b edges
gmon.out: No such file or directory
usuario@debian:~/Descargas/lab$ gprof -l edges
gmon.out: No such file or directory
usuario@debian:~/Descargas/lab$ gcc -o edges edges.c -Wl,--no-as-needed -lprofiler
usuario@debian:~/Descargas/lab$ CUPROFILE=/tmp/edges.prof ./edges img.pgm out.pgm
PROFILE: interrupts/evictions/bytes = 206/116/9136
usuario@debian:~/Descargas/lab$ google-pprof --text edges /tmp/edges.prof
Using local file edges.
Using local file /tmp/edges.prof.
Removing killpg from all stack traces.
Total: 206 samples
142 68.9% 68.9%      142 68.9% gaussian
58 28.2% 97.1%      58 28.2% laplacian
3 1.5% 98.5%        3 1.5% fputc
1 0.5% 99.0%        1 0.5% _close
1 0.5% 99.5%        1 0.5% load_image_file
1 0.5% 100.0%       5 2.4% save_image_file
0 0.0% 100.0%       1 0.5% _IO_file_close_it
0 0.0% 100.0%      206 100.0% __libc_start_main
0 0.0% 100.0%      206 100.0% _start
0 0.0% 100.0%      200 97.1% edges
0 0.0% 100.0%       1 0.5% fclose
0 0.0% 100.0%      206 100.0% main
usuario@debian:~/Descargas/lab$
```

Ejemplo: Perfil plano

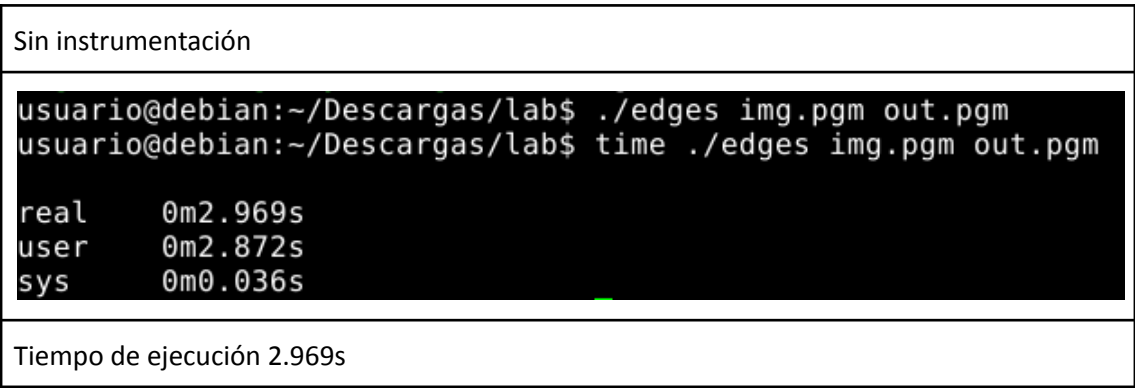
```
$ gcc -o bucles bucles.c -Wl,--no-as-needed -lprofiler
$ CUPROFILE=/tmp/prof CUPROFILE_FREQUENCY=200 ./bucles
PROFILE: interrupts/evictions/bytes = 272/131/10232
$ google-pprof --text bucles /tmp/prof
Total: 65 samples
28 43.1% 43.1%      28 43.1% atan
24 36.9% 80.0%      24 36.9% producto
12 18.5% 98.5%      12 18.5% division
1 1.5% 100.0%       29 44.6% atangente
0 0.0% 100.0%       65 100.0% __libc_start_main
0 0.0% 100.0%       65 100.0% _start
0 0.0% 100.0%       65 100.0% main
```

- Número de muestras
- Porcentaje
- Porcentaje acumulado
- Número de muestras incl. llamadas
- Porcentaje incluyendo llamadas

Genera un grafo de llamadas en formato SVG y visualízalo en un navegador web con:
\$ google-pprof --web edges /tmp/edges.prof
También se puede generar una imagen en formato PDF (--pdf), GIF (--gif), SVG (--svg)...



Obtén el tiempo de ejecución del programa sin instrumentación, con la instrumentación por defecto y con instrumentación en tiempo real con frecuencia 1000. Calcula la sobrecarga producida por la instrumentación en ambos casos.



Instrumentación por defecto

```
usuario@debian:~/Descargas/lab$ CPUPROFILE=/tmp/edges.prof time ./edges img.pgm out.pgm
2.87user 0.02system 0:02.92elapsed 99%CPU (0avgtext+0avgdata 16948maxresident)k
0inputs+12296outputs (0major+4275minor)pagefaults 0swaps
```

Tiempo de ejecución 2.92s

Sobrecarga respecto instrumentación en tiempo real=1,6%

Con instrumentación en tiempo real con frecuencia

```
usuario@debian:~/Descargas/lab$ CPUPROFILE=/tmp/edges.prof CPUPROFILE_FREQUENCY=1000 CPUPROFILE_REAL
ME=1 time ./edges img.pgm out.pgm
2.76user 0.05system 0:02.85elapsed 98%CPU (0avgtext+0avgdata 16948maxresident)k
0inputs+12296outputs (0major+4275minor)pagefaults 0swaps
```

Tiempo de ejecución 2.85s

Sobrecarga respecto Sin instrumentación=2,4%

Obtén un perfil de ejecución por líneas de código añadiendo la opción --lines. Es necesario haber compilado el programa con la opción -g.

Compara gprof y google-pprof en cuanto a implementación, sobrecarga, precisión, exactitud, ámbito de medida, resolución temporal y facilidad de uso.

```
usuario@debian:~/Descargas/lab$ gcc -g -o edges edges.c -Wl,--no-as-needed -lprofiler
usuario@debian:~/Descargas/lab$ CPUPROFILE=/tmp/edges.prof ./edges img.pgm out.pgm
usuario@debian:~/Descargas/lab$ google-pprof --text --lines edges /tmp/edges.prof
```

```
usuario@debian:~/Descargas/lab$ google-pprof --text --lines edges /tmp/edges.prof
Using local file edges.
Using local file /tmp/edges.prof.
Removing killpg from all stack traces.
Total: 274 samples
179 65.3% 65.3%      183 66.8% gaussian /home/usuario/Descargas/lab/edges.c:126
 52 19.0% 84.3%      53 19.3% laplacian /home/usuario/Descargas/lab/edges.c:166
  8  2.9% 87.2%       8  2.9% _IO_getc ???:0
  6  2.2% 89.4%      15  5.5% gaussian /home/usuario/Descargas/lab/edges.c:125
  5  1.8% 91.2%       7  2.6% gaussian /home/usuario/Descargas/lab/edges.c:129
  5  1.8% 93.1%       7  2.6% save_image_file /home/usuario/Descargas/lab/edges.c:88
  3  1.1% 94.2%       6  2.2% gaussian /home/usuario/Descargas/lab/edges.c:124
  3  1.1% 95.3%       3  1.1% load_image_file /home/usuario/Descargas/lab/edges.c:65
  2  0.7% 96.0%       2  0.7% fputc ???:0
  2  0.7% 96.7%       2  0.7% gaussian /home/usuario/Descargas/lab/edges.c:122
  2  0.7% 97.4%       2  0.7% laplacian /home/usuario/Descargas/lab/edges.c:164
  2  0.7% 98.2%       2  0.7% laplacian /home/usuario/Descargas/lab/edges.c:171
  2  0.7% 98.9%      13  4.7% load_image_file /home/usuario/Descargas/lab/edges.c:66
  1  0.4% 99.3%       5  1.8% laplacian /home/usuario/Descargas/lab/edges.c:165
  1  0.4% 99.6%       1  0.4% laplacian /home/usuario/Descargas/lab/edges.c:169
  1  0.4% 100.0%      1  0.4% save_image_file /home/usuario/Descargas/lab/edges.c:87
  0  0.0% 100.0%     274 100.0% __libc_start_main ???:0
  0  0.0% 100.0%     274 100.0% _start ???:0
  0  0.0% 100.0%      96 35.0% edges /home/usuario/Descargas/lab/edges.c:181
  0  0.0% 100.0%     99 36.1% edges /home/usuario/Descargas/lab/edges.c:182
  0  0.0% 100.0%     58 21.2% edges /home/usuario/Descargas/lab/edges.c:183
  0  0.0% 100.0%       1  0.4% laplacian /home/usuario/Descargas/lab/edges.c:170
  0  0.0% 100.0%     13  4.7% main /home/usuario/Descargas/lab/edges.c:195
  0  0.0% 100.0%    253 92.3% main /home/usuario/Descargas/lab/edges.c:197
  0  0.0% 100.0%       8  2.9% main /home/usuario/Descargas/lab/edges.c:199
```

.google-pprof: Añade instrumentación con una biblioteca dinámica. Más avanzado que gcc/gprof.
google-pprof está menos limitado que gprof, ya que con google-pprof podemos cambiar la frecuencia y nos puede mostrar:
Tiempo Real, hilos y bibliotecas de enlace dinámico

gcc/gprof: Añade instrumentación en la compilación para generar perfiles de ejecución. Mientras que gprof, solo muestra el tiempo de procesador en modo usuario y el hilo principal. No muestra las bibliotecas de enlace dinámico

```
usuario@debian:~/Escritorio/laboratorio$ gprof -l edges
Flat profile:

Each sample counts as 0.01 seconds.
%   cumulative   self           self         total
time  seconds    seconds   calls   Ts/call   Ts/call   name
50.55      1.21      1.21           3      0.00     0.00  laplacian (edges.c:157 @ 400f38)
17.13      1.62      0.41           1      0.00     0.00  laplacian (edges.c:155 @ 400f5a)
16.71      2.03      0.40           1      0.00     0.00  __libc_csu_init
5.22       2.15      0.13           1      0.00     0.00  laplacian (edges.c:153 @ 400eee)
2.72       2.22      0.07           1      0.00     0.00  laplacian (edges.c:151 @ 400f10)
2.09       2.27      0.05           1      0.00     0.00  main (edges.c:199 @ 401170)
0.84       2.29      0.02           1      0.00     0.00  _fini
0.84       2.31      0.02           1      0.00     0.00  laplacian (edges.c:152 @ 400ed5)
0.84       2.33      0.02           1      0.00     0.00  laplacian (edges.c:164 @ 400f8a)
0.63       2.34      0.02           1      0.00     0.00  laplacian (edges.c:155 @ 400f1f)
0.63       2.36      0.02           1      0.00     0.00  laplacian (edges.c:156 @ 400f28)
0.42       2.37      0.01           1      0.00     0.00  gaussian (edges.c:101 @ 400b98)
0.42       2.38      0.01           1      0.00     0.00  gaussian (edges.c:121 @ 400ce4)
0.42       2.39      0.01           1      0.00     0.00  laplacian (edges.c:163 @ 400f81)
0.42       2.40      0.01           1      0.00     0.00  laplacian (edges.c:165 @ 400f96)
0.42       2.41      0.01           1      0.00     0.00  main (edges.c:202 @ 401193)
0.00       2.41      0.00           3      0.00     0.00  gaussian (edges.c:100 @ 400b76)
0.00       2.41      0.00           1      0.00     0.00  laplacian (edges.c:140 @ 400dfb)
0.00       2.41      0.00           1      0.00     0.00  load_image_file (edges.c:18 @ 40081c)
```