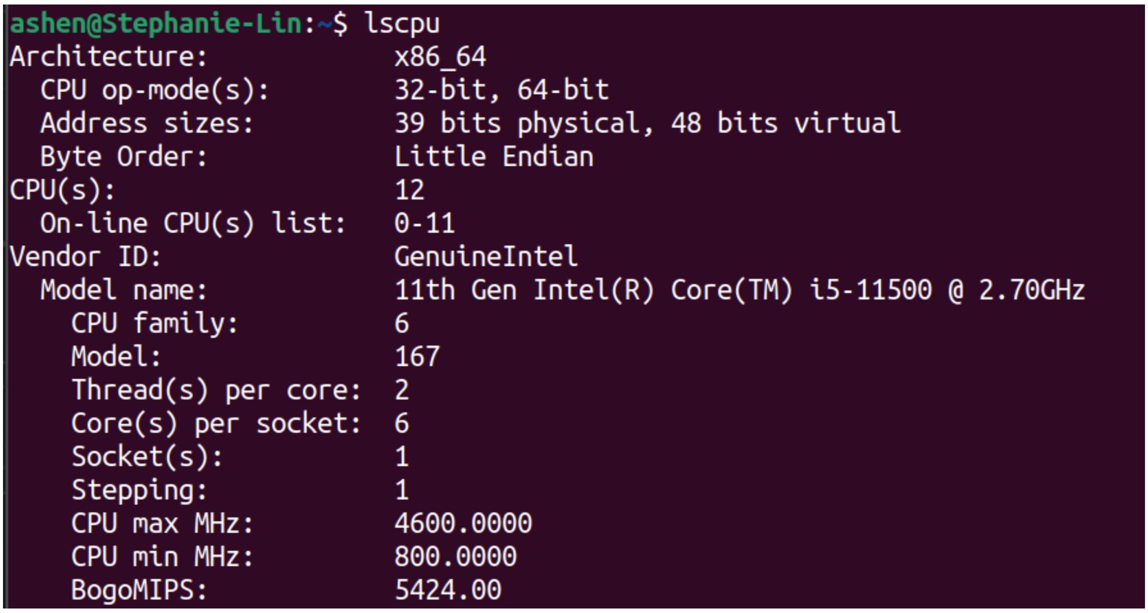
軟體分析與最佳化 HW1

612410017 林靖紳

Execute the dhrystone benchmark

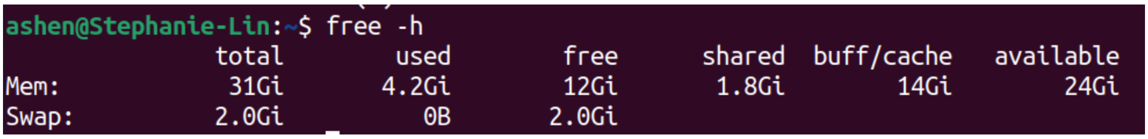
Execution environments

- CPU information



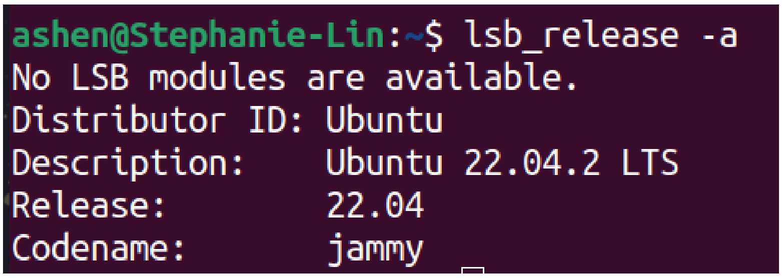
```
ashen@Stephanie-Lin:~$ lscpu
Architecture:            x86_64
  CPU op-mode(s):        32-bit, 64-bit
  Address sizes:         39 bits physical, 48 bits virtual
  Byte Order:            Little Endian
CPU(s):                  12
  On-line CPU(s) list:   0-11
Vendor ID:               GenuineIntel
  Model name:            11th Gen Intel(R) Core(TM) i5-11500 @ 2.70GHz
    CPU family:          6
    Model:               167
    Thread(s) per core:  2
    Core(s) per socket:  6
    Socket(s):           1
    Stepping:            1
    CPU max MHz:         4600.0000
    CPU min MHz:         800.0000
    BogoMIPS:            5424.00
```

- Memory



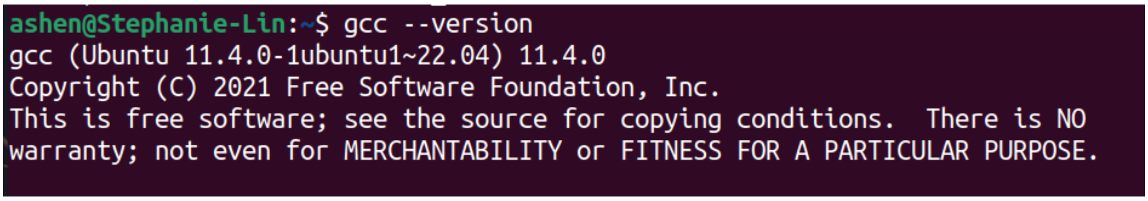
```
ashen@Stephanie-Lin:~$ free -h
               total        used        free      shared  buff/cache   available
Mem:            31Gi       4.2Gi        12Gi       1.8Gi        14Gi        24Gi
Swap:          2.0Gi          0B       2.0Gi
```

- OS version



```
ashen@Stephanie-Lin:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 22.04.2 LTS
Release:        22.04
Codename:       jammy
```

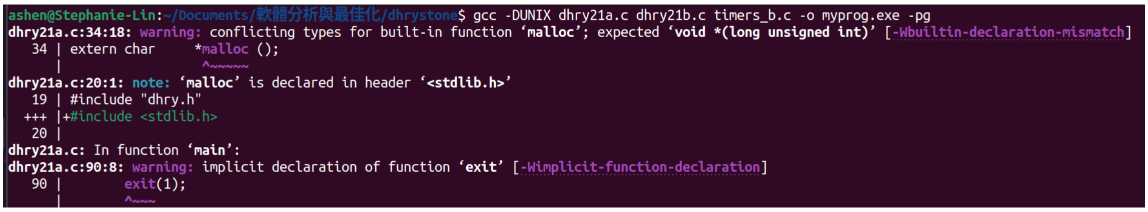
- GCC version



```
ashen@Stephanie-Lin:~$ gcc --version
gcc (Ubuntu 11.4.0-1ubuntu1~22.04) 11.4.0
Copyright (C) 2021 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

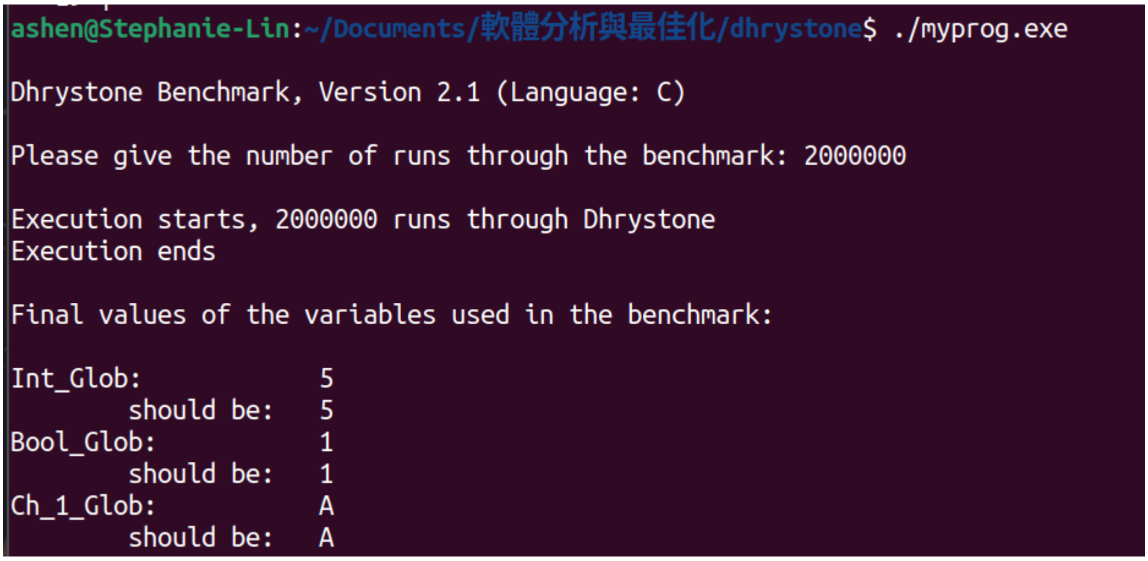
Execute the benchmark step by step

1. Compile Program



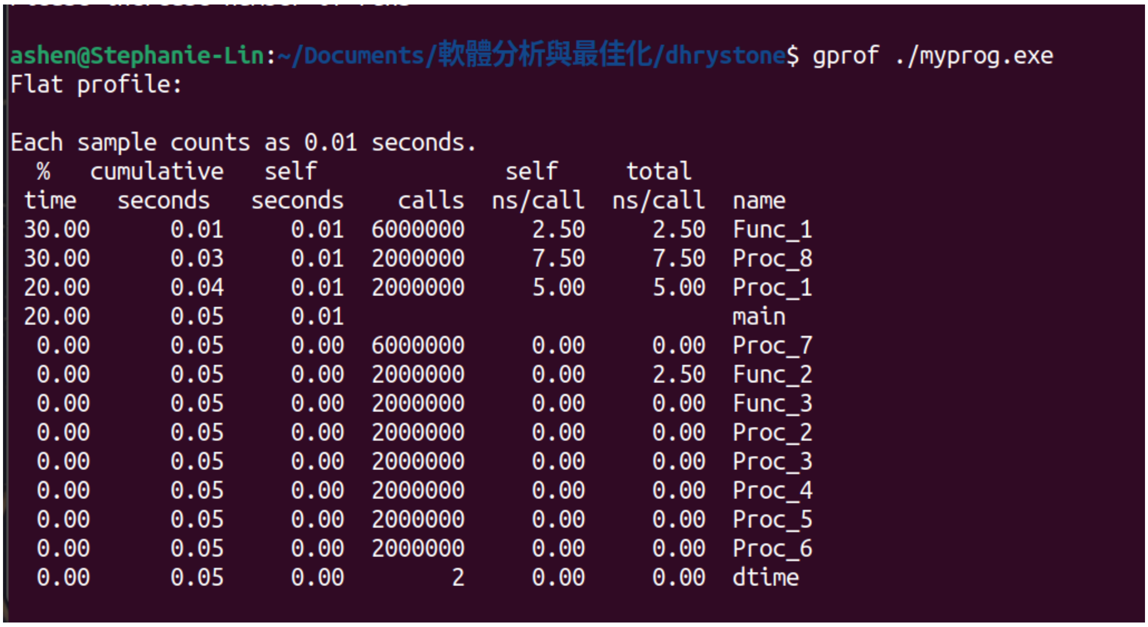
```
ashen@Stephanie-Lin:~/Documents/軟體分析與最佳化/dhrystone$ gcc -DUNIX dhry21a.c dhry21b.c timers_b.c -o myprog.exe -pg
dhry21a.c:34:18: warning: conflicting types for built-in function ‘malloc’; expected ‘void *(long unsigned int)’ [-Wbuiltin-declaration-mismatch]
   34 | extern char     *malloc ();
      |                  ^~~~~~
dhry21a.c:20:1: note: ‘malloc’ is declared in header ‘<stdlib.h>’
   19 | #include "dhry.h"
  +++ |+#include <stdlib.h>
   20 |
dhry21a.c: In function ‘main’:
dhry21a.c:90:8: warning: implicit declaration of function ‘exit’ [-Wimplicit-function-declaration]
   90 |        exit(1);
      |        ^~~~
```

2. Execute with number of runs = 2000000



```
ashen@Stephanie-Lin:~/Documents/軟體分析與最佳化/dhrystone$ ./myprog.exe

Dhrystone Benchmark, Version 2.1 (Language: C)

Please give the number of runs through the benchmark: 2000000

Execution starts, 2000000 runs through Dhrystone
Execution ends

Final values of the variables used in the benchmark:

Int_Glob:             5
        should be:    5
Bool_Glob:            1
        should be:    1
Ch_1_Glob:            A
        should be:    A
```

3. Profiling the program

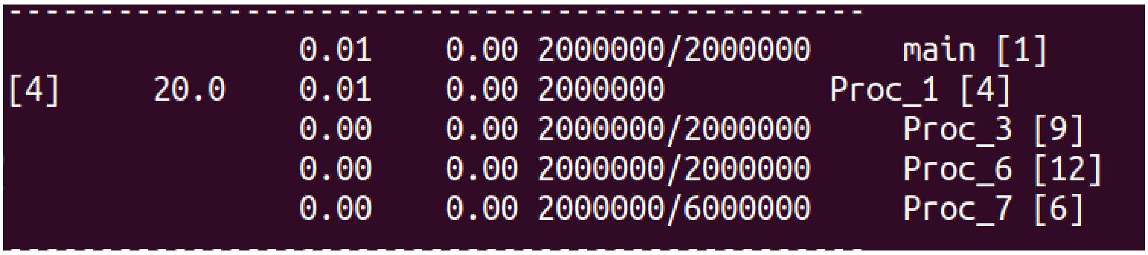


```
ashen@Stephanie-Lin:~/Documents/軟體分析與最佳化/dhrystone$ gprof ./myprog.exe
Flat profile:

Each sample counts as 0.01 seconds.
  %   cumulative   self              self     total
 time   seconds   seconds    calls  ns/call  ns/call  name
 30.00      0.01     0.01  6000000     2.50     2.50  Func_1
 30.00      0.03     0.01  2000000     7.50     7.50  Proc_8
 20.00      0.04     0.01  2000000     5.00     5.00  Proc_1
 20.00      0.05     0.01                             main
  0.00      0.05     0.00  6000000     0.00     0.00  Proc_7
  0.00      0.05     0.00  2000000     0.00     2.50  Func_2
  0.00      0.05     0.00  2000000     0.00     0.00  Func_3
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_2
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_3
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_4
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_5
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_6
  0.00      0.05     0.00        2     0.00     0.00  dtime
```

Answer

Q1: Which functions are called by Proc_1 ?

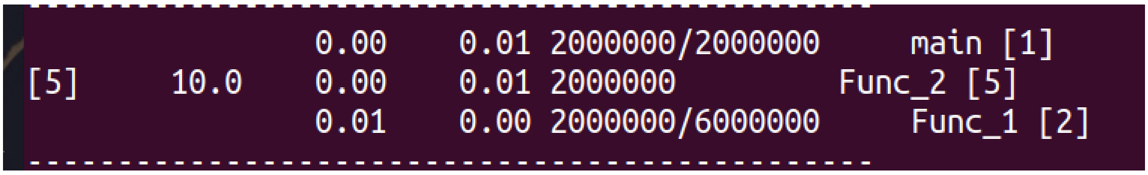
- Proc_3, Proc_6, Proc_7



```
-----------------------------------------------
                0.01    0.00 2000000/2000000     main [1]
[4]     20.0    0.01    0.00 2000000         Proc_1 [4]
                0.00    0.00 2000000/2000000     Proc_3 [9]
                0.00    0.00 2000000/2000000     Proc_6 [12]
                0.00    0.00 2000000/6000000     Proc_7 [6]
-----------------------------------------------
```

Q2: Which functions are called by Func_2?

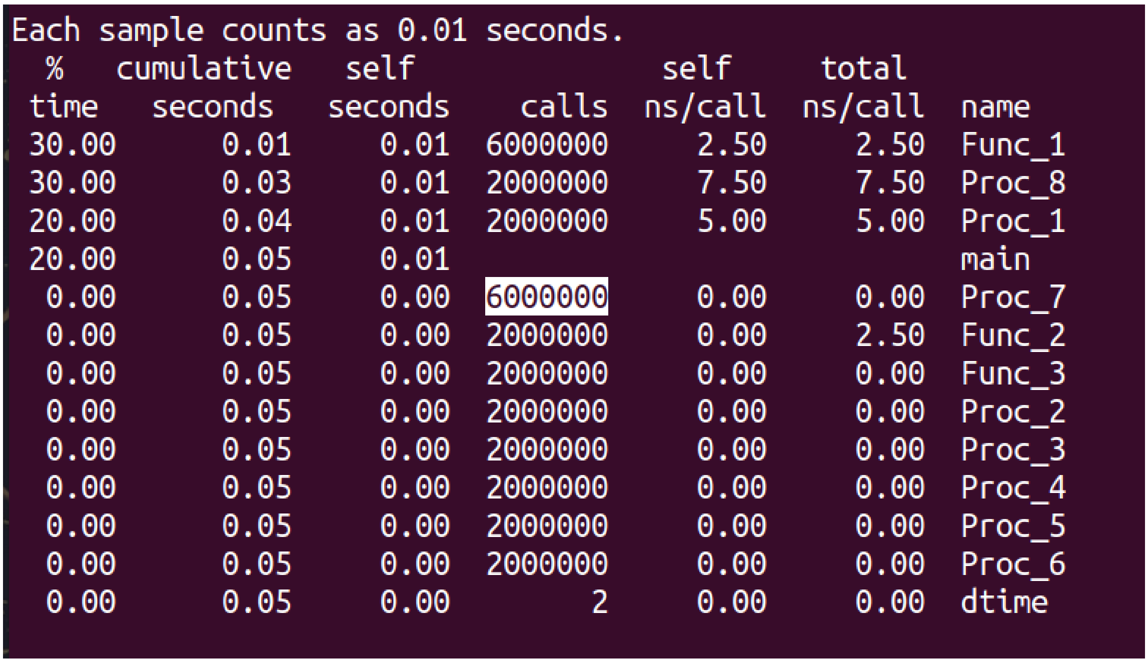
- Func_1



```
-----------------------------------------------
                0.00    0.01 2000000/2000000     main [1]
[5]     10.0    0.00    0.01 2000000         Func_2 [5]
                0.01    0.00 2000000/6000000     Func_1 [2]
-----------------------------------------------
```

Q3: How many times are Proc_3 and Proc_7 called separately?

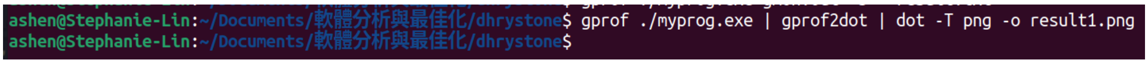
- Proc_3: 2000000
- Proc_7: 6000000



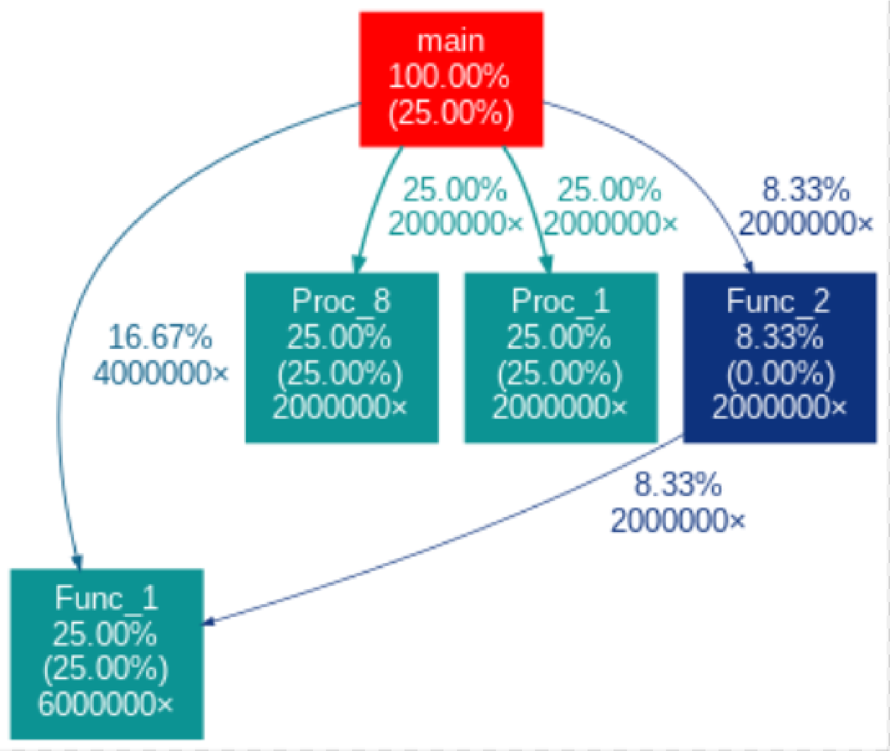
```
Each sample counts as 0.01 seconds.
  %   cumulative   self              self     total
 time   seconds   seconds    calls  ns/call  ns/call  name
 30.00      0.01     0.01  6000000     2.50     2.50  Func_1
 30.00      0.03     0.01  2000000     7.50     7.50  Proc_8
 20.00      0.04     0.01  2000000     5.00     5.00  Proc_1
 20.00      0.05     0.01                             main
  0.00      0.05     0.00  6000000     0.00     0.00  Proc_7
  0.00      0.05     0.00  2000000     0.00     2.50  Func_2
  0.00      0.05     0.00  2000000     0.00     0.00  Func_3
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_2
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_3
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_4
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_5
  0.00      0.05     0.00  2000000     0.00     0.00  Proc_6
  0.00      0.05     0.00        2     0.00     0.00  dtime
```

Q4: Use gprof2dot (https://github.com/jrfonseca/gprof2dot (https://github.com/jrfonseca/gprof2dot)) and Graphviz (https://graphviz.org/ (https://graphviz.org/)) to convert gprof text information to a corresponding PNG file.

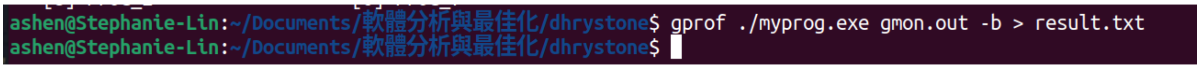
- gprog2dot



```
ashen@Stephanie-Lin:~/Documents/軟體分析與最佳化/dhrystone$ gprof ./myprog.exe | gprof2dot | dot -T png -o result1.png
ashen@Stephanie-Lin:~/Documents/軟體分析與最佳化/dhrystone$
```



將 $ gprof ./myprog.exe 的執行結果轉存之純文字檔。



```
ashen@Stephanie-Lin:~/Documents/軟體分析與最佳化/dhrystone$ gprof ./myprog.exe gmon.out -b > result.txt
ashen@Stephanie-Lin:~/Documents/軟體分析與最佳化/dhrystone$ 
```