Eric Paulz

ECE 4930-004

**HW3**

***Part 1: Installing FlipIt***

[epaulz@login001 matmul]$ ./long

Fault injector seed: 533

Starting faulty computation.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Start\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Successfully injected Integer Data error!!

Rank: 0

Total # faults injected: 1

Bit position is: 27

Index of the fault site: 240

Fault site probability: 1.000000e-08

Chosen random probability is: 9.794167e-09

Attempts since last injection: 8113437

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*End\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Number of incorrect elements: 1

[epaulz@login001 matmul]$

***Part 2: Compile HPCCG***

[epaulz@login001 HPCCG-1.0]$ mpirun -n 8 ./test\_HPCCG 32 32 32

Initial Residual = 1944.57

Iteration = 15 Residual = 18.1609

Iteration = 30 Residual = 0.043237

Iteration = 45 Residual = 0.000160905

Iteration = 60 Residual = 7.41837e-07

Iteration = 75 Residual = 3.07385e-09

Iteration = 90 Residual = 1.42189e-11

Iteration = 105 Residual = 5.99938e-14

Iteration = 120 Residual = 2.62093e-16

Iteration = 135 Residual = 5.8031e-19

Iteration = 149 Residual = 1.20172e-21

Mini-Application Name: hpccg

Mini-Application Version: 1.0

Parallelism:

Number of MPI ranks: 8

OpenMP not enabled:

Dimensions:

nx: 32

ny: 32

nz: 32

Number of iterations: : 149

Final residual: : 1.20172e-21

\*\*\*\*\*\*\*\*\*\* Performance Summary (times in sec) \*\*\*\*\*\*\*\*\*\*\*:

Time Summary:

Total : 0.849358

DDOT : 0.12653

WAXPBY : 0.0734024

SPARSEMV: 0.644499

FLOPS Summary:

Total : 2.49981e+09

DDOT : 1.56238e+08

WAXPBY : 2.34357e+08

SPARSEMV: 2.10921e+09

MFLOPS Summary:

Total : 2943.17

DDOT : 1234.79

WAXPBY : 3192.77

SPARSEMV: 3272.63

DDOT Timing Variations:

Min DDOT MPI\_Allreduce time: 0.0127676

Max DDOT MPI\_Allreduce time: 0.0651364

Avg DDOT MPI\_Allreduce time: 0.0318385

SPARSEMV OVERHEADS:

SPARSEMV MFLOPS W OVERHEAD: 3121.47

SPARSEMV PARALLEL OVERHEAD Time: 0.0312111

SPARSEMV PARALLEL OVERHEAD Pct: 4.61901

SPARSEMV PARALLEL OVERHEAD Setup Time: 0.0270848

SPARSEMV PARALLEL OVERHEAD Setup Pct: 4.00835

SPARSEMV PARALLEL OVERHEAD Bdry Exch Time: 0.00412631

SPARSEMV PARALLEL OVERHEAD Bdry Exch Pct: 0.610663

[epaulz@login001 HPCCG-1.0]$

***Part 3: Instrument HPCCG for Fault Injection***

[epaulz@login001 HPCCG-1.0]$ mpirun -n 8 ./test\_HPCCG 32 32 32 --stateFile /home/epaulz/FlipIt//.HPCCG --numberFaulty 1 --faulty 2 29634

Fault injector seed: 233

Initial Residual = 1944.57

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Start\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Successfully injected 64-bit IEEE Float Data error!!

Rank: 2

Total # faults injected: 1

Bit position is: 47

Index of the fault site: 84509

Fault site probability: 1.000000e-08

Chosen random probability is: 4.862908e-09

Attempts since last injection: 292195571

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*End\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Iteration = 45 Residual = 0.000160905

Mini-Application Name: hpccg

Mini-Application Version: 1.0

Parallelism:

Number of MPI ranks: 8

OpenMP not enabled:

Dimensions:

nx: 32

ny: 32

nz: 32

Number of iterations: : 85

Final residual: : 8.73231e-11

\*\*\*\*\*\*\*\*\*\* Performance Summary (times in sec) \*\*\*\*\*\*\*\*\*\*\*:

Time Summary:

Total : 28.686

DDOT : 12.4724

WAXPBY : 1.50439

SPARSEMV: 14.6818

FLOPS Summary:

Total : 1.42606e+09

DDOT : 8.9129e+07

WAXPBY : 1.33693e+08

SPARSEMV: 1.20324e+09

MFLOPS Summary:

Total : 49.7129

DDOT : 7.14612

WAXPBY : 88.8686

SPARSEMV: 81.9546

DDOT Timing Variations:

Min DDOT MPI\_Allreduce time: 0.0298896

Max DDOT MPI\_Allreduce time: 11.6615

Avg DDOT MPI\_Allreduce time: 10.0234

SPARSEMV OVERHEADS:

SPARSEMV MFLOPS W OVERHEAD: 78.7424

SPARSEMV PARALLEL OVERHEAD Time: 0.598935

SPARSEMV PARALLEL OVERHEAD Pct: 3.91955

SPARSEMV PARALLEL OVERHEAD Setup Time: 0.571971

SPARSEMV PARALLEL OVERHEAD Setup Pct: 3.74309

SPARSEMV PARALLEL OVERHEAD Bdry Exch Time: 0.0269642

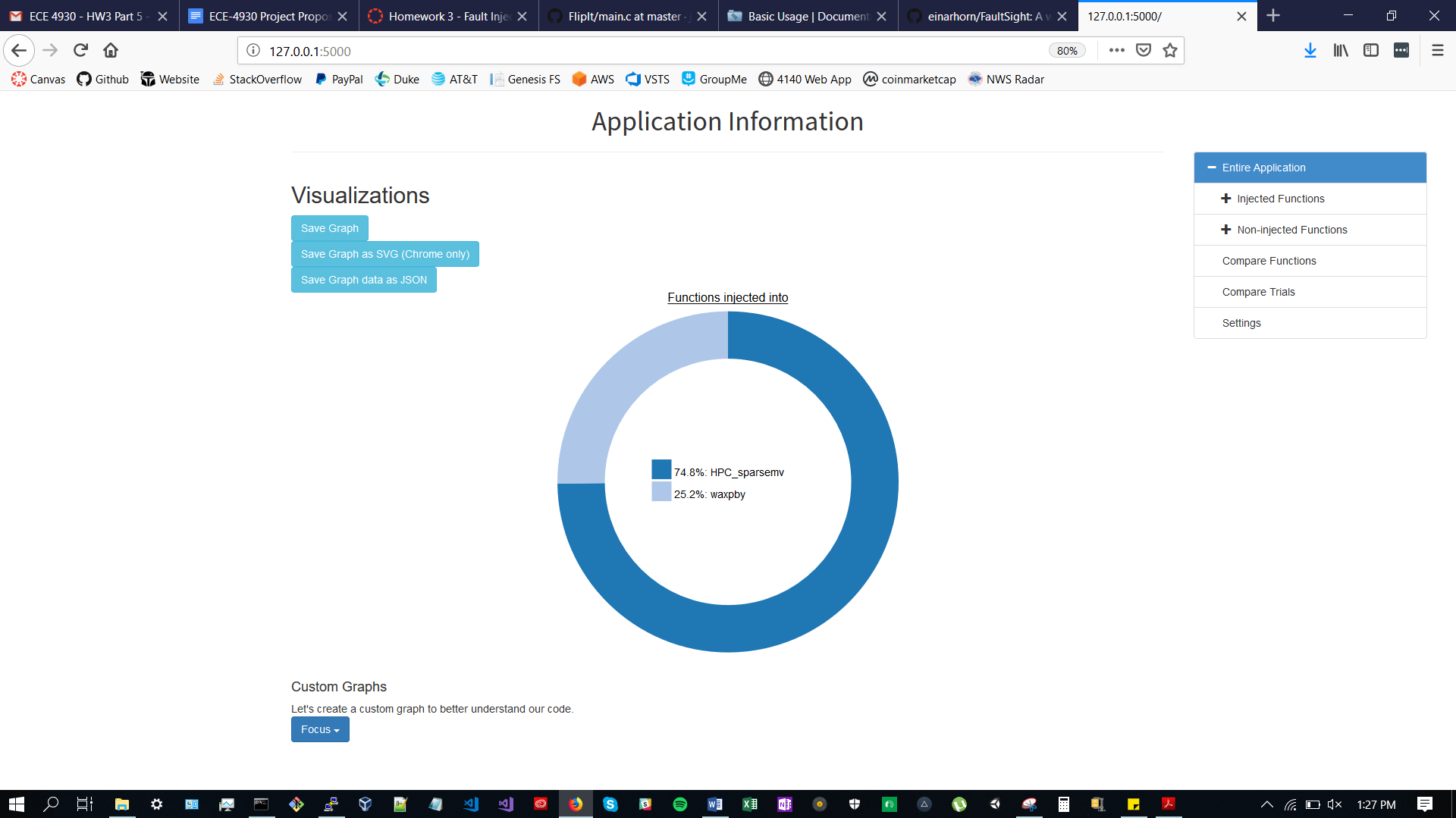
SPARSEMV PARALLEL OVERHEAD Bdry Exch Pct: 0.176459

[epaulz@login001 HPCCG-1.0]$

***Part 4: Conduct a Fault Injection Campaign***

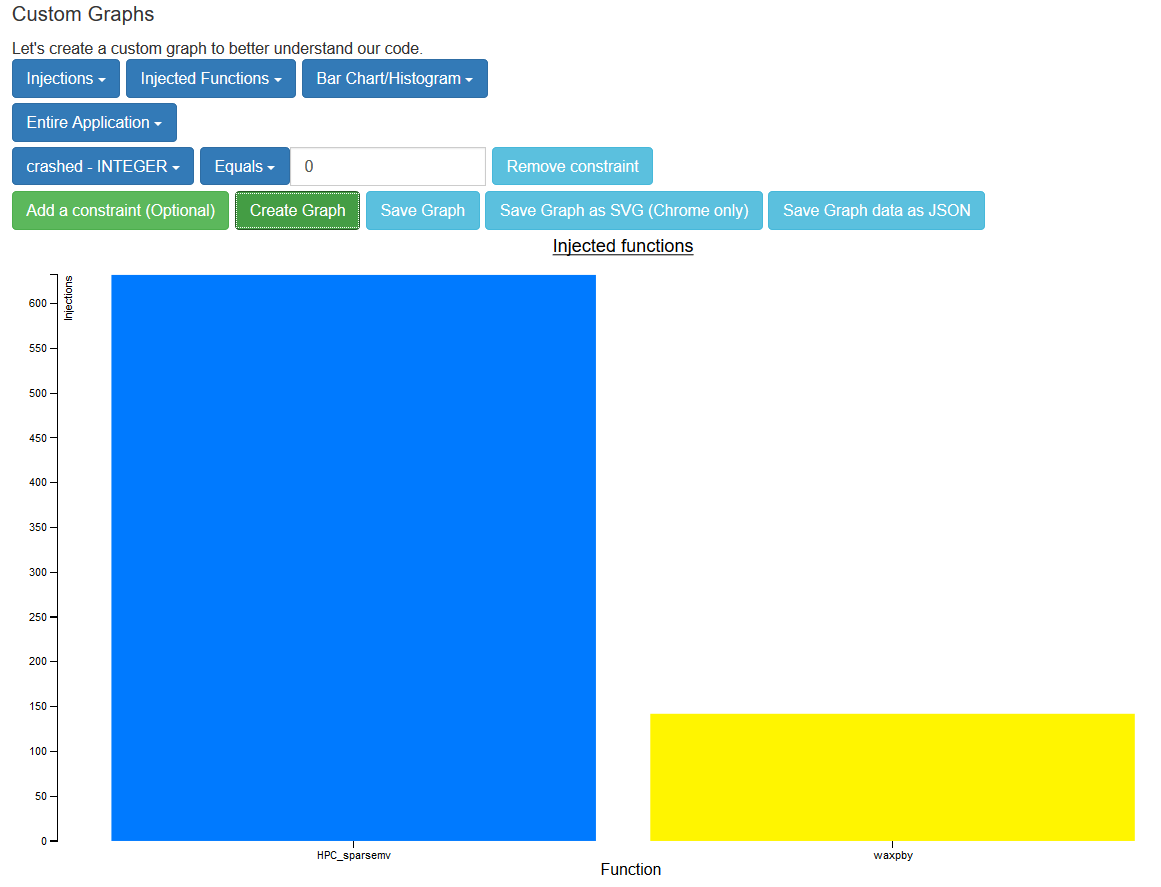
--- done ---

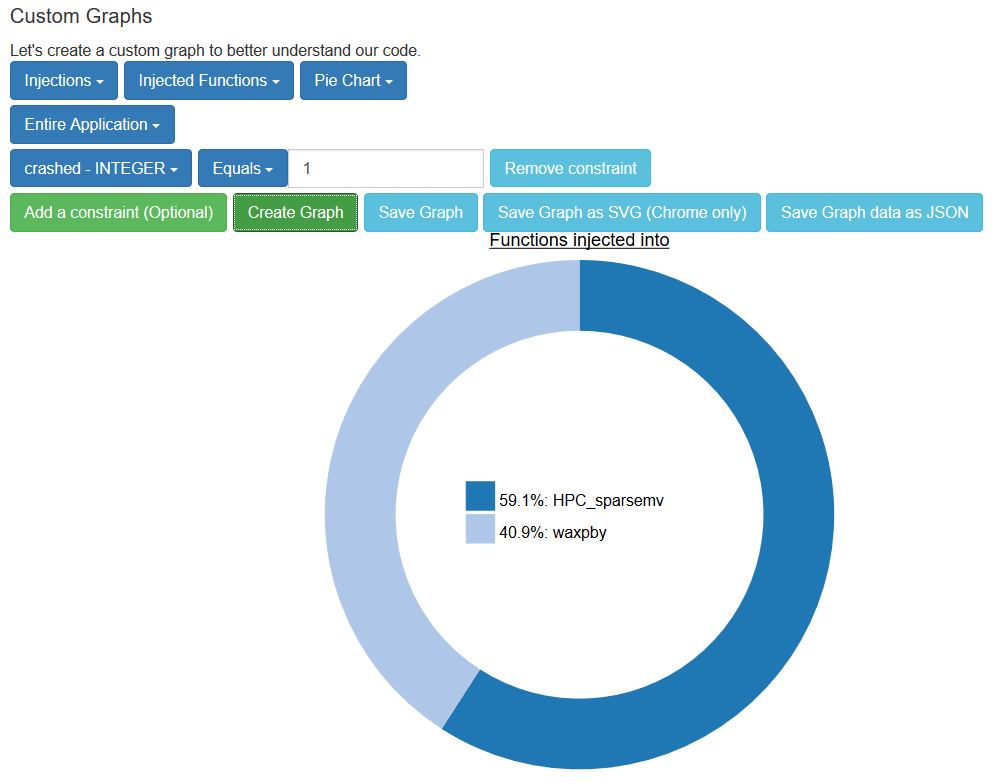
***Part 5: Installing FaultSight***



***Part 6: Fault Injection Analysis with FaultSight***

* What are the functions that faults are injected into?
  + HPC\_sparsemv
  + waxpby
* What is the percentage of fault injection trials that have detection?
  + 0%
* What are the bit locations that have detection?
  + 12-23
  + 24-35
  + 36-47
  + 48-63
* What is the percent of trials that segfaulted?
  + Can’t figure out how to see segfaults in FaultSight.
* What are the bit locations that generate a segfault? Why do you think these bits cause a segfault?
  + Same as above.
* What are the bit locations in ddot where we inject faults?
  + Empty dataset returned?
* What line of code suffers the most injections? Why do you think this line suffers the most injections?
  + HPC\_sparsemv
    - Line 74
  + waxpby
    - Line 57
* Use the custom constraints to generate two plots that you find interesting. Explain why you find these plots interesting and provide an analysis of what the plots show.





These graphs are interesting because they show the difference between which trials crashed and which ones did not among the injections in these two functions.

Based on the above analysis does anything stand out to you? Finally, estimate how, if at all, the added error detector could improve the application’s reliability.

Not really.

It could improve reliability by showing the user or developer where they can make improvements. The GUI functionality of FaultSight is helpful for visualizing the data.