Lab-report

ECPE 170– Computer Systems and Networks – Fall 2015

Name: Zhiyun Yan

Lab Topic: Performance Optimization (Lab #: 6)

(1) What is the total physical RAM installed in the system? (In MB)

Answer: 8142mb installed in the system

(2) With no applications running (beyond the web browser with this page), how much RAM is used by the native operating system? (e.g. Windows)

Answer: 3653mb is used by the native operating system.

(3) With no applications running (beyond the web browser with this page), how much RAM is available?

Answer: 4724mb is available.

(4) Check the virtual machine configuration. How much RAM is currently allocated to Linux in your virtual machine?

Note: Your answer to question 4 must be less than your answer to question 3! Otherwise, your system will use slow virtual memory (i.e. swapping data to the hard disk) when running this lab.

Answer: 1024mb is currently allocated to Linux in my virtual machine.

(5) Try to increase your virtual machine memory allocation, if possible, to the maximum allowed based on your free RAM. Leave ~256MB free for the virtual machine program itself. Now how much RAM is allocated to Linux in your virtual machine?

Answer: I increased the ram to 2048mb.

(6) Boot Linux. With no applications running in Linux, how much RAM is available *inside* the virtual machine? The "System Monitor" program should report that information. This is the space that is actually available for our test

application.

Answer:1449mb is available inside the virtual machine.

7) What is the code doing? (Describe the algorithm in a paragraph, focusing on the combine1() function.)

Answer: This function is used for adding elements to the array. The elements data is float type. The vector length is time of running. If the vector length is 0, this function will return, else the elements will be into the vector.

(8) What is the largest number of elements that the vector can hold WITHOUT using swap storage (virtual memory), and how much memory does it take? Be sure to leave enough memory for **Firefox** and**LibreOffice**, since you'll need those when running this lab as well.

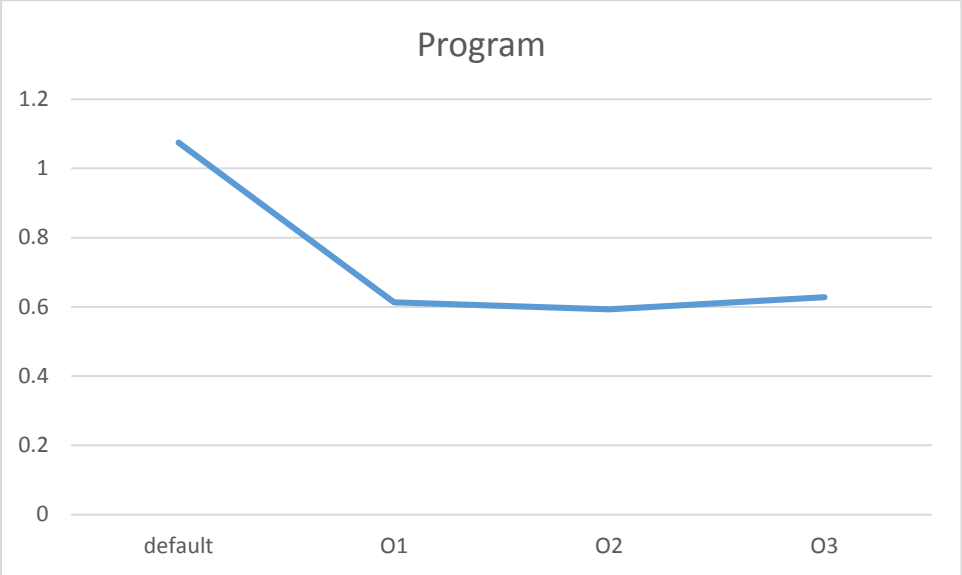
Answer: The largest number of elements that the vector can hold is 380314961, it takes 1450.79mb

(9) What vector size are you using for all experiments in this lab?

Answer: the vector size I used is 100,000,000

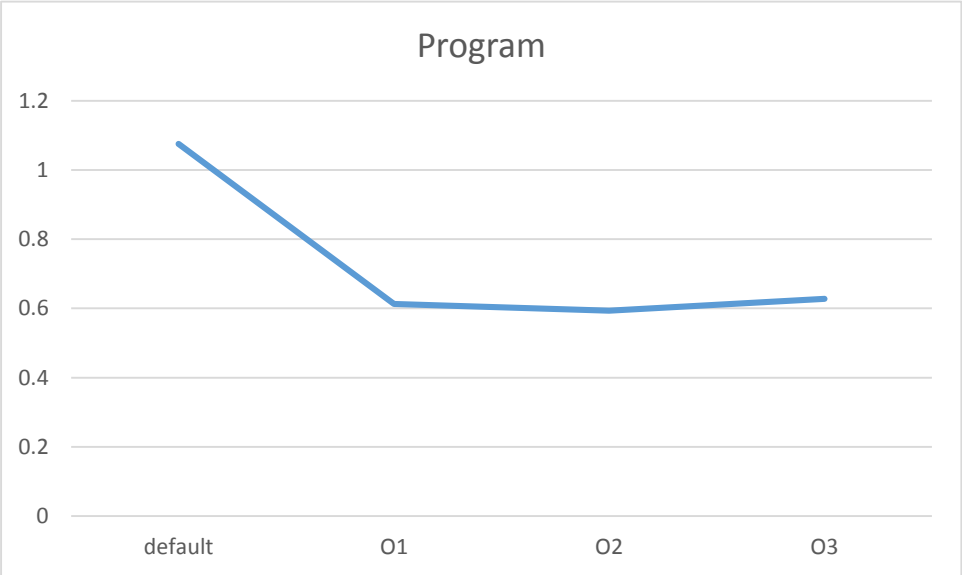
(10) How much time does the **compiler** take to finish with (a) no optimization, (b) with -O1 optimization, (c) with -O2 optimization, and (d) with -O3 optimization? Report the Real time, which is the "wall clock" time. Create both a table and a graph in LibreOffice Calc.

	default	O1	O2	O3
Complier	0.120s	0.186s	0.197s	0.204s



(11) How much time does the **program** take to finish with (a) no optimization, (b) with -O1 optimization, (c) with -O2 optimization, and (d) with -O3 optimization? Report the Real time, which is the "wall clock" time. Create both a table and a graph in LibreOffice Calc.

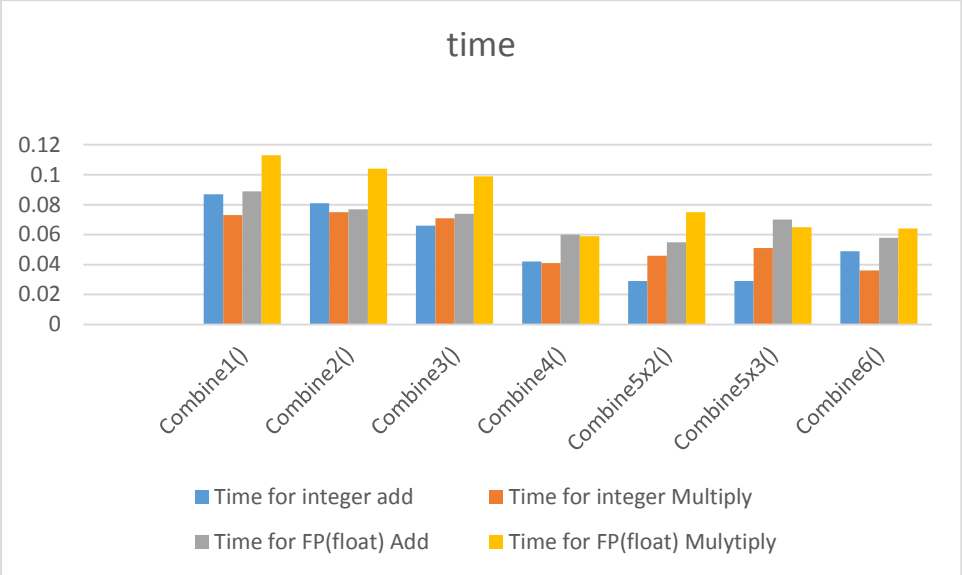
	default	O1	O2	O3
Program	1.075s	0.613s	0.593s	0.628s



(12) After implementing each function, benchmark it for a variety of data types and mathematical operations. Fill in the table below as you write each function.

Configuration	Vector Size (element)	Vector Size (MB)	Time for integer add	Time for integer Multiply	Time for FP(float) Add	Time for FP(float) Mulytiply
Combine1()	100000000	381.47	0.087	0.073	0.089	0.113
Combine2()	100000000	381.47	0.081	0.075	0.077	0.104
Combine3()	100000000	381.47	0.066	0.071	0.074	0.099
Combine4()	100000000	381.47	0.042	0.041	0.060	0.059
Combine5x2()	100000000	381.47	0.029	0.046	0.055	0.075
Combine5x3()	100000000	381.47	0.029	0.051	0.070	0.065
Combine6()	100000000	381.47	0.049	0.036	0.058	0.064

(13) Using LibreOffice Calc, create a single graph that shows the data in the table created, specifically the four time columns. (You don't need to plot vector size)



(14) As a reminder, you should be using version control to track your code, and ensure that the final code is checked in along with your report PDF.