

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ

по лабораторной работе №3

по дисциплине «Качество и метрология программного обеспечения»

**Тема: «Измерение характеристик динамической сложности программ с
помощью профилировщика SAMPLER»**

Студент гр. 7304

Овчинников Н.В.

Преподаватель

Кирияничков В.А.

Санкт-Петербург

2021

Задание

1. Ознакомиться с документацией на монитор SAMPLER и выполнить под его управлением тестовые программы `test_cyc.c` и `test_sub.c` с анализом параметров повторения циклов, структуры описания циклов, способов профилирования процедур и проверкой их влияния на точность и чувствительность профилирования.

2. Скомпилировать и выполнить под управлением SAMPLER'a программу на С, разработанную в 1-ой лабораторной работе.

Выполнить разбиение программы на функциональные участки и снять профили для двух режимов: 1 - измерение только полного времени выполнения программы; 2 - измерение времен выполнения функциональных участков (ФУ).

Убедиться, что сумма времен выполнения ФУ соответствует полному времени выполнения программы.

3. Выявить "узкие места", связанные с ухудшением производительности программы, ввести в программу усовершенствования и получить новые профили. Объяснить смысл введенных модификаций программ.

Ход работы

Выполняется **вариант 14**

Использовался **старый SAMPLER**, т.к. новый не работает на используемой виртуальной машине с *Microsoft Windows XP Professional 2002 Service Pack 3*.

Для компиляции программ использовался *Borland C++ 3.1*.

Профилирование тестовой программы **TEST_CYC.CPP**.

Список обработанных файлов.

NN	Имя обработанного файла
1.	TEST_CYC.CPP

Таблица с результатами измерений (используется 13 из 416 записей)

Исх.Поз.	Прием.Поз.	Общее время(мкс)	Кол-во прох.	Среднее время(мкс)
1 : 8	1 : 10	4335.47	1	4335.47
1 : 10	1 : 12	8670.11	1	8670.11
1 : 12	1 : 14	21671.50	1	21671.50
1 : 14	1 : 16	43348.03	1	43348.03
1 : 16	1 : 19	4334.64	1	4334.64
1 : 19	1 : 22	8670.11	1	8670.11
1 : 22	1 : 25	21677.37	1	21677.37
1 : 25	1 : 28	43343.00	1	43343.00
1 : 28	1 : 34	4339.66	1	4339.66
1 : 34	1 : 40	8670.11	1	8670.11
: 40	1 : 46	21671.50	1	21671.50
1 : 46	1 : 52	43348.03	1	43348.03

В результате профилирования видно, что полученные данные сильно завышены, что может объясняться накладными расходами эмулятора.

В программе используется разная запись циклов с одинаковым количеством итераций. По результатам профилирования заметно, что это не влияет на результат.

Также видна линейная зависимость времени выполнения от количества итераций.

Профилирование тестовой программы **TEST_SUB.CPP**.

Список обработанных файлов.

NN	Имя обработанного файла
1.	TEST_SUB.CPP

Таблица с результатами измерений (используется 5 из 416 записей)

Исх.Поз.	Прием.Поз.	Общее время(мкс)	Кол-во прох.	Среднее время(мкс)
1 : 30	1 : 32	433700.70	1	433700.70
1 : 32	1 : 34	867387.15	1	867387.15
1 : 34	1 : 36	2168474.16	1	2168474.16
1 : 36	1 : 38	4336937.43	1	4336937.43

По результатам профилирования можно сделать аналогичные выводы о завышении результатов из-за использования эмулятора и линейной зависимости между временем выполнения цикла и количеством итераций в нём.

Профилирование полного времени выполнения программы из первой лабораторной работы (**LINFIT_FULL.CPP**).

Список обработанных файлов.

NN	Имя обработанного файла
1.	LINFIT~1.CPP

Таблица с результатами измерений (используется 2 из 416 записей)

Исх.Поз.	Прием.Поз.	Общее время(мкс)	Кол-во прох.	Среднее время(мкс)
1 : 47	1 : 56	57681.15	1	57681.15

Общее время выполнения программы **57681 мкс.**

Профилирование времени выполнения функциональных участков программы из первой лабораторной работы (**LINFIT_FUN.CPP**).

Список обработанных файлов.

NN	Имя обработанного файла
1.	LINFIT~4.CPP

Таблица с результатами измерений (используется 14 из 416 записей)

Исх.Поз.	Прием.Поз.	Общее время(мкс)	Кол-во прох.	Среднее время(мкс)
1 :	8 1 : 19	4.19	1	4.19
:	19 1 : 22	2.51	1	2.51
1 :	22 1 : 34	17765.97	1000	17.77
1 :	34 1 : 22	1754.97	999	1.76
1 :	34 1 : 37	2.51	1	2.51
1 :	37 1 : 45	470.17	1	470.17
1 :	45 1 : 48	1.68	1	1.68
1 :	48 1 : 52	8196.58	1000	8.20
1 :	52 1 : 48	1765.87	999	1.77
1 :	52 1 : 55	1.68	1	1.68

1 : 55	1 : 81	2.51	1	2.51
1 : 65	1 : 69	2.51	1	2.51
1 : 69	1 : 74	30869.61	1000	30.87
1 : 74	1 : 69	1758.33	999	1.76
1 : 74	1 : 77	1.68	1	1.68
1 : 77	1 : 8	7.54	1	7.54

Сумма времени выполнения функциональных участков почти совпадает с полным временем выполнения программы из предыдущего пункта (с разницей **5 мс**), что можно объяснить погрешностями и накладными расходами монитора за счёт большего количества измерений.

Больше всего времени уходит на циклы в строках **22:34, 48:52 и 69:74**.

Время в цикле **48:52** можно сократить за счёт уменьшения операций косвенного обращения к памяти (разыменования указателей *a* и *b*) с помощью использования дополнительных локальных переменных.

В цикле **69:74** можно убрать операции деления с остатком и вычитания, т.к. требуется заполнить массивы случайными данными и неважно из какого диапазона на результаты профилирования это не повлияет.

Цикл **22:34** можно было бы ускорить за счёт использования векторных инструкций, но они не доступны в такой ранней версии компилятора.

Профилирование полного времени выполнения оптимизированной версии программы из первой лабораторной работы (**LINFIT_FULL_OPT.CPP**).

Список обработанных файлов. -----	
NN	Имя обработанного файла
1.	LINFIT~3.CPP

Таблица с результатами измерений (используется 2 из 416 записей)

Исх.Поз.	Прием.Поз.	Общее время(мкс)	Кол-во прох.	Среднее время(мкс)
1 : 50	1 : 59	54684.12	1	54684.12

Общее время выполнения программы **54684 мкс**, что на **3 мс** быстрее предыдущего варианта.

Профилирование времени выполнения функциональных участков оптимизированной версии программы из первой лабораторной работы (**LINFIT_FUN_OPT.CPP**).

Список обработанных файлов.

NN	Имя обработанного файла
1.	LINFIT~2.CPP

Таблица с результатами измерений (используется 15 из 416 записей)

Исх.Поз.	Прием.Поз.	Общее время(мкс)	Кол-во прох.	Среднее время(мкс)
1 : 8	1 : 19	3.35	1	3.35
1 : 19	1 : 22	2.51	1	2.51
1 : 22	1 : 34	17771.84	1000	17.77
1 : 34	1 : 22	1783.47	999	1.79
1 : 34	1 : 37	2.51	1	2.51
1 : 37	1 : 45	468.50	1	468.50
: 45	1 : 48	2.51	1	2.51
1 : 48	1 : 52	7500.13	1000	7.50
1 : 52	1 : 48	1762.52	999	1.76
1 : 52	1 : 55	2.51	1	2.51
1 : 55	1 : 60	4.19	1	4.19
1 : 60	1 : 85	2.51	1	2.51
1 : 70	1 : 73	2.51	1	2.51

1 : 73	1 : 78	28219.55	1000	28.22
1 : 78	1 : 73	1765.03	999	1.77
1 : 78	1 : 81	2.51	1	2.51
1 : 81	1 : 8	6.70	1	6.70

Сумма времени выполнения функциональных участков почти совпадает с полным временем выполнения программы из предыдущего пункта (с разницей **5 мс**), что можно объяснить погрешностями и накладными расходами монитора за счёт большего количества измерений.

Среднее время одной итерации в цикле *48:52* уменьшилось на **0.3 мкс**.

Среднее время одной итерации в цикле *69:74* уменьшилось на **2.65 мкс**.

Выводы

В ходе выполнения лабораторной работы был изучен монитор для профилирования **SAMPLER**. С его помощью было выполнено профилирование тестовых программ *TEST_CYC.CPP* и *TEST_SUB.CPP*. В результате было установлено, что время выполнения цикла не зависит от его записи и линейно зависит от количества итераций.

Были получены профили программы из первой лабораторной работы в двух режимах: полное время выполнения и время выполнения функциональных участков. В результате были выявлены узкие места и выполнены оптимизации. После чего повторно были получены профили в двух режимах. В результате применения оптимизаций среднее время выполнения одной итерации в цикле 48:52 уменьшилось на **0.3 мкс**, в цикле 69:74 уменьшилось на **2.65 мкс**.