

Московский авиационный институт
(национальный исследовательский университет)

Факультет информационных технологий и прикладной
математики

Кафедра вычислительной математики и программирования

Лабораторная работа №3 по курсу «Дискретный анализ»

Студент: К. М. Воронов
Преподаватель: А. А. Кухтичев
Группа: М8О-207Б-19
Дата:
Оценка:
Подпись:

Москва, 2020

Лабораторная работа №3

Задача: Для реализации словаря из предыдущей лабораторной работы необходимо провести исследование скорости выполнения и потребления оперативной памяти. В случае выявления ошибок или явных недочётов, требуется их исправить.
Используемые утилиты: valgrind, gprof.

1 Valgrind

Как сказано в [1]: «Valgrind — инструментальное программное обеспечение, предназначенное для отладки использования памяти, обнаружения утечек памяти, а также профилирования.».

С его помощью я устранял утечки памяти в своей программе. Valgrind показывает, в какой функции произошла ошибка, связанная с памятью. В итоге ошибок с памятью в моей программе нет.

```
==28917== Memcheck, a memory error detector
==28917== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==28917== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==28917== Command: ./solution
==28917==
==28917==
==28917== HEAP SUMMARY:
==28917==       in use at exit: 0 bytes in 0 blocks
==28917==    total heap usage: 333,313 allocs, 333,313 frees, 98,736,856 bytes
allocated
==28917==
==28917== All heap blocks were freed --no leaks are possible
==28917==
==28917== For lists of detected and suppressed errors, rerun with: -s
==28917== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```

2 Gprof

Как сказано в [2], Gprof - средство профилирования в Unix системах. Используется для измерения времени работы отдельных функций программы и общего времени работы программы. Профилировщик показывает, сколько процентов от общего времени работы программы работает функция, сколько раз вызывается, и многое другое. Диагностика проводилась на тесте из 10^6 строк с запросами на поиск, добавление и удаление.

```
kirill@kirill-VirtualBox:~/Рабочий стол/DA/lab2$ gprof ./solution -p ./gmon.out
Flat profile:
```

Each sample counts as 0.01 seconds.

%	cumulative	self	self	total		name
time	seconds	seconds	calls	ms/call	ms/call	
31.24	0.73	0.73	333019	0.00	0.00	Search(TPatricia*&,TString&)
23.54	1.28	0.55	333311	0.00	0.00	Add(TPatricia*&,unsigned long long,TString&)
20.54	1.76	0.48	333670	0.00	0.00	Patricia_delete(TPatricia*&,TStr
6.85	1.92	0.16	76481916	0.00	0.00	TString::operator[](int)
6.42	2.07	0.15	18914912	0.00	0.00	Number(TString&,int)
4.71	2.18	0.11				main
3.00	2.25	0.07	1	70.10	70.10	TPatricia::~~TPatricia()
1.71	2.29	0.04	666329	0.00	0.00	TString::operator==(TString const&)
0.86	2.31	0.02	1000000	0.00	0.00	TString::Clear()
0.43	2.32	0.01	333311	0.00	0.00	TString::operator=(TString const&)
0.43	2.33	0.01	333310	0.00	0.00	Searchnumber(TString&,TString&)
0.43	2.34	0.01	1	10.01	10.01	_GLOBAL__sub_I__ZlRK7TStringS1_
0.00	2.34	0.00	333670	0.00	0.00	TString::operator!=(TString const&)
0.00	2.34	0.00	333311	0.00	0.00	TString::TString()
0.00	2.34	0.00	333311	0.00	0.00	TPatricia::TPatricia()
0.00	2.34	0.00	333019	0.00	0.00	TString::Sizev()
0.00	2.34	0.00	1	0.00	0.00	__static_initialization_and_destru

Как видно, из-за необходимости поиска в удалении и добавлении, эти функции занимают немалую часть времени работы программы.

3 Дневник отладки

1. 05.11.2020 Изучил утилиту Valgrind.
2. С 05.11.2020 начал пользоваться Valgrind, устранять утечки и ошибки.
3. 27.11.2020 Изучил принцип работы с утилитой Gprof
4. С 27.11.2020 Производил профилирование программы с помощью Gprof

4 Выводы

Выполнив третью лабораторную работу по курсу «Дискретный анализ», я научился пользоваться Valgrind и Gprof, устранять ошибки связанные с памятью, анализировать время работы программы, ускорять её. Всё это помогло сделать мою программу лучше.

Список литературы

[1] *Valgrind* - *Wikipedia*.

URL: <https://ru.wikipedia.org/wiki/Valgrind> (дата обращения 10.11.2020)

[2] *Gprof* - *Wikipedia*.

URL: <https://en.wikipedia.org/wiki/Gprof> (дата обращения: 27.11.2020).