

Министерство образования и науки Российской Федерации
Санкт-Петербургский государственный электротехнический университет
«ЛЭТИ»
Факультет компьютерных технологий и информатики
Кафедра вычислительной техники

Отчёт по лабораторной работе №1
«Работа с множествами»
по дисциплине
«Алгоритмы и структуры данных»

Вариант 23

Выполнили студенты
факультета КТИ группы №3305
Лоуцкер Алексей и Григорьева Анна

Проверил старший преподаватель
Колинко Павел Георгиевич

Оглавление

1. Задание.....	3
2. Формализация задания.....	3
3. Контрольные тесты.....	4
4. Временная сложность.....	5
4.1. Ожидаемая асимптотическая:.....	5
4.2 Фактическая:.....	5
5. Результаты измерения времени обработки каждым из способов.....	6
6. Выводы.....	7
7. Список источников.....	7
8. Исходные тексты.....	7

1. Задание

(Вариант 23)

Универсум: Шестнадцатеричные цифры

Формулировка задания: Множество, содержащее цифры из A, не встречающиеся в B и не являющиеся общими для C и D.

2. Формализация задания

$E = A \ \& \ (\sim B) \ \& \ (\sim (C \ \& \ D))$

3. Контрольные тесты

В качестве контрольного теста мы использовали множества следующих размеров и содержания:

A = 123ABCDEF

B = BCED45678

C = ABC23456

D = 9876532

В результате мы получили новое множество E:

E = 1AF

Полученное нами новое множество соответствует результатам, полученным по средствам ручной проверки, что подтверждает верную работу программы, по крайней мере, для данного теста.

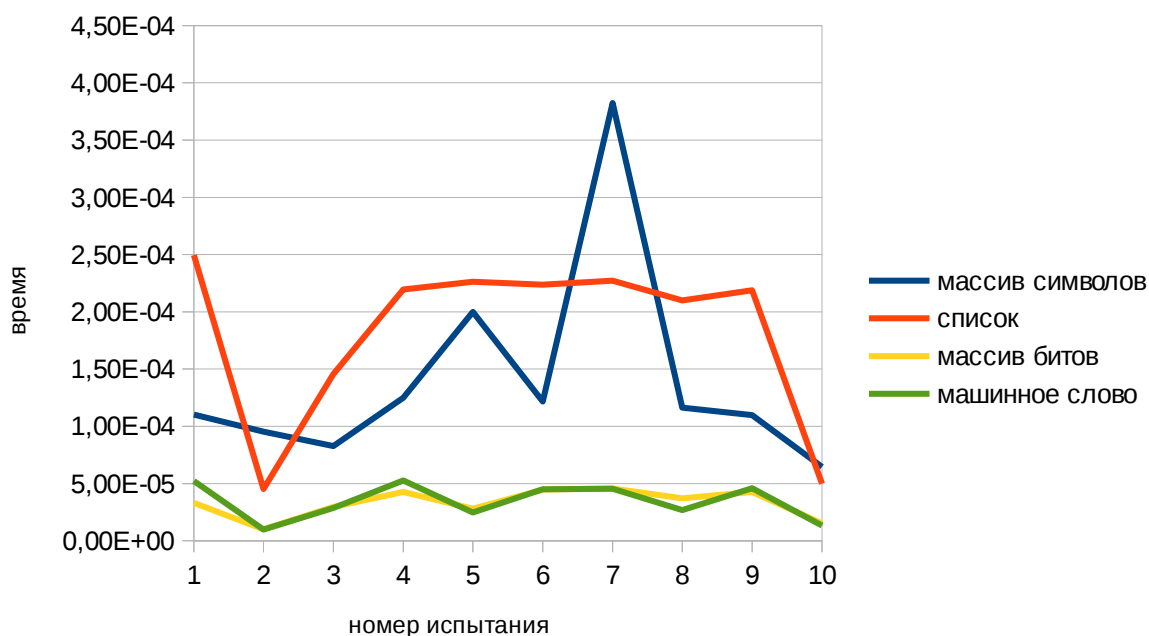
4. Временная сложность

4.1. Ожидаемая асимптотическая:

	1 — массив символов	2 — список символов	3 — массив битов	4 — машинное слово
Set() - конструктор по умолчанию /создание случайного множества/	$O(n)$	$O(n)$	$O(u)$	$O(1)$
Set(char*) - конструктор множества из строки	$O(n)$	$O(n)$	$O(u)$	$O(u)$
Set(Set&) - конструктор копий	$O(n)$	$O(n)$	$O(u)$	$O(1)$
operator= - присваивание	$O(n)$	$O(n)$	$O(u)$	$O(1)$
operator~ - отрицание	$O(n^2)$	$O(n^2)$	$O(u)$	$O(1)$
operator& - пересечение	$O(n^2)$	$O(n^2)$	$O(u)$	$O(1)$
$E = A \& \sim B \& \sim (C \& D)$	$O(n^2)$	$O(n^2)$	$O(u)$	$O(1)$

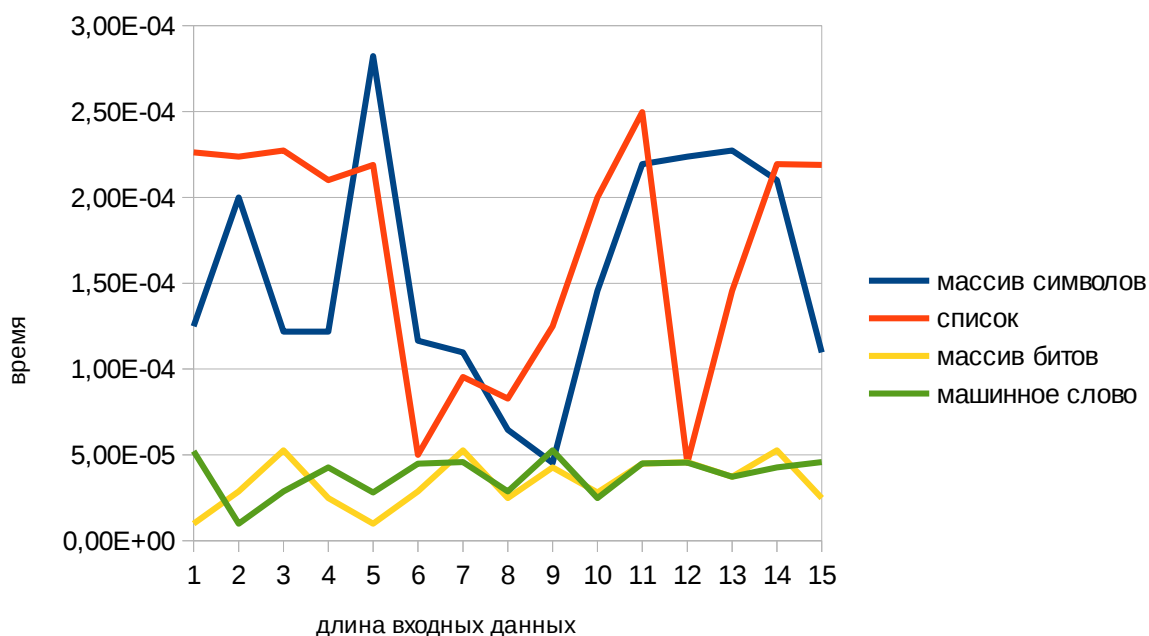
n — размер конкретного множества, u — размер универсума, $n \leq u$

4.2 Фактическая:



5. Результаты измерения времени обработки каждым из способов

На случайно сгенерированных примерах множеств были получены значения времени при различных размерах входных данных



Можно заметить, что:

- время решения задачи способами (3) и (4) не зависит от содержания входных данных; это соответствует таблице 4.1

- время обработки способами (1) и (2) не коррелирует с размерами множеств и количеством совершаемых операций над ними, из чего следует вывод, что на время обработки каждого из способов большее влияние оказывают запущенные фоновые процессы, нежели особенности каждого из тестов.

6. Выводы

Опираясь на таблицу и график, представленные в пункте «временная сложность», можно говорить о том, что варианты представления множеств в качестве массива битов(3) и машинного слова(4) показывают лучшее время работы, используя при этом минимальное количество памяти. Однако, при необходимости расширения универсума или добавления дополнительного функционала, они потребуют значительной доработки или вовсе окажутся непригодными для использования.

Варианты представления множества в качестве массива символов(1) или односвязного списка(2), несмотря на большее время работы, имеют свои преимущества: в (1) - случайный доступ к элементу множества, а в (2) - добавление или удаление случайного элемента за $O(1)$. Также, в случае использования массива или односвязного списка, мы имеем возможность легко расширить функционал, что позволяет нам использовать эти структуры данных для гораздо более широкого круга задач.

7. Список источников

1) Алгоритмы и структуры данных: методические указания к лабораторным работам, практическим занятиям и курсовому проектированию / сост. П. Г. Колинко. - СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2013

2) <http://stackoverflow.com>

3) <http://cplusplus.com>

8. Исходные тексты

github.com/alout1/4sets