Міністерство освіти і науки України
Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"
Фізико-технічний інститут

Дисципліна “Спеціальні розділи обчислювальної математики”
Комп’ютерний практикум

Робота №1. Багаторозрядна арифметика

Виконав
студент гр. ФБ-11 Подолянко Т.О.
Перевірив
Грубіян Є.О.

Київ — 2023

Робота №1. Багаторозрядна арифметика

Мета роботи: Отримання практичних навичок програмної реалізації багаторозрядної арифметики; ознайомлення з прийомами ефективної реалізації критичних по часу ділянок програмного коду та методами оцінки їх ефективності.

Завдання до комп'ютерного практикуму згідно з варіантом №13

А) Згідно варіанту розробити клас чи бібліотеку функцій для роботи з m-бітними цілими числами. Бібліотека повинна підтримувати числа довжини до 2048 біт. Повинні бути реалізовані такі операції:

1) переведення малих констант у формат великого числа (зокрема, 0 та 1);

2) додавання чисел;

3) віднімання чисел;

4) множення чисел, піднесення чисел до квадрату;

5) ділення чисел, знаходження остачі від ділення;

6) піднесення числа до багаторозрядного степеня;

7) конвертування (переведення) числа в символьну строку та обернене перетворення символьної строки у число; обов’язкова підтримка шістнадцяткового представлення, бажана – десяткового та двійкового.

Бажано реалізувати такі операції:

1) визначення номеру старшого ненульового біта числа;

2) бітові зсуви (вправо та вліво), які відповідають діленню та множенню на степені двійки.

Мова програмування, семантика функцій та спосіб реалізації можуть обиратись довільним чином.

Хід роботи

1. Програмна реалізація алгоритмів багаторозрядної арифметики

Завдання практикуму реалізовано мовою програмування Rust. Вихідні коди розміщено у інтернет-репозиторії GitHub за посиланням: https://github.com/timofey282228/sprzom-lab1_2.git.

Реалізовано всі обов’язкові операції, а також бітовий зсув вліво, вправо та визначення номеру старшого ненульового біта числа.

Для контролю коректності роботи алгоритмів реалізовані відповідні юніт-тести, зокрема, для $(a+b)\cdot c = ac + bc$ (equality_1), $n \cdot a = a + a + .. + a$ (equality_2) для деяких обраних багаторозрядних чисел. Контрольні результати отримані за допомогою наданого онлайн-ресурсу: https://srom-check.herokuapp.com/calculate-long-arithmetic.

```
PS>cargo test --lib
    Finished test [unoptimized + debuginfo] target(s) in 0.00s
     Running unittests src\lib.rs (target\debug\deps\vl_big_ints-c88d5d9862ef1816.exe)

running 13 tests
test tests::add_test ... ok
test tests::div_test ... ok
test tests::mul_test ... ok
test tests::get_bit_test ... ok
test tests::equality_1 ... ok
test tests::sub_test ... ok
test tests::pow_test ... ok
test tests::shl_test ... ok
test tests::ordering_test ... ok
test tests::get_highest_set_bit_test ... ok
test tests::shr_test ... ok
test tests::mul_single_digit_test ... ok
test tests::equality_2 ... ok

test result: ok. 13 passed; 0 failed; 0 ignored; 0 measured; 0 filtered out; finished in 0.00s
```

2. Оцінка середнього часу виконання операцій

Для вимірювання середнього часу виконання операцій реалізовано програму, що виконує базові операції (додавання, множення, віднімання, ділення) на випадкових даних різних довжин та усереднює результат за певною кількістю експериментів. Оцінки середнбого часу наводяться у стандартному виводі, а також зберігаються у CSV у форматі (операція, к-ть експериментів, довжина операндів (біт), час виконання (нс)).

Приклад виконання:

```
PS>cargo run --example performance --release
    Finished release [optimized] target(s) in 0.01s
     Running `target\release\examples\performance.exe`
Performance calculation example

Operand length: 1024 bits

Experiment count: 1000
Running 1000 experiments on 1024-bit bigints; operation: +
 - Average operaition duration: 174ns
Running 1000 experiments on 1024-bit bigints; operation: -
 - Average operaition duration: 77ns
Running 1000 experiments on 1024-bit bigints; operation: *
 - Average operaition duration: 5.697µs
Running 1000 experiments on 1024-bit bigints; operation: /
 - Average operaition duration: 496ns

Experiment count: 10000
Running 10000 experiments on 1024-bit bigints; operation: +
 - Average operaition duration: 173ns
Running 10000 experiments on 1024-bit bigints; operation: -
 - Average operaition duration: 102ns
Running 10000 experiments on 1024-bit bigints; operation: *
 - Average operaition duration: 4.799µs
Running 10000 experiments on 1024-bit bigints; operation: /
 - Average operaition duration: 501ns

Experiment count: 100000
Running 100000 experiments on 1024-bit bigints; operation: +
 - Average operaition duration: 179ns
Running 100000 experiments on 1024-bit bigints; operation: -
 - Average operaition duration: 112ns
Running 100000 experiments on 1024-bit bigints; operation: *
 - Average operaition duration: 4.667µs
Running 100000 experiments on 1024-bit bigints; operation: /
 - Average operaition duration: 493ns

Operand length: 2048 bits
```

Приклад результату роботи:

Операція	**К-ть вимірів**	**Довжина операндів, біт**	**Середній час, нс**
+	1000	1024	174
+	1000	2048	293
+	1000	4096	395
+	10000	1024	173
+	10000	2048	261
+	10000	4096	433
+	100000	1024	179
+	100000	2048	260
+	100000	4096	397
/	1000	1024	496
/	1000	2048	583
/	1000	4096	907
/	10000	1024	501
/	10000	2048	602

/	10000	4096	897
/	100000	1024	493
/	100000	2048	634
/	100000	4096	886
*	1000	1024	5697
*	1000	2048	16646
*	1000	4096	68016
*	10000	1024	4799
*	10000	2048	26514
*	10000	4096	88868
*	100000	1024	4667
*	100000	2048	28044
*	100000	4096	87732
-	1000	1024	77
-	1000	2048	137
-	1000	4096	166
-	10000	1024	102
-	10000	2048	142
-	10000	4096	225
-	100000	1024	112
-	100000	2048	168
-	100000	4096	235

Якщо на отриманих даних побудувати графіки, то можна побачити, що залежність часу виконання від довжини входу близька до лінійної.

Також слід зазначити, що значні витрати часу на множення, і порівняно незначні — на ділення пояснюються використанням найменш ефективної реалізації множення та діленням чисел, близьких за розміром.