Міністерство освіти України

Київський Національний Університет імені Тараса Шевченка

Механіко-математичний факультет

Кафедра алгебри і комп’ютерної математики

КУРСОВА РОБОТА на тему:

«Стандарт числа POSIT. Потенційна заміна стандарту IEEE 754»

Виконав студент 4 курсу,  
групи Комп’ютерна математика 2  
Тищенко Тимофій Андрійович

Науковий керівник: Бородін Віктор Анатолійович

Київ – 2023

**Зміст**

1. Вступ .................................................................................................................. 3
2. Основи типу числа POSIT ................................................................................. 3
3. Відмінності між POSIT та IEEE 754 ................................................................... 4
4. Принцип роботи типу числа POSIT ................................................................. 4
5. Переваги та недоліки POSIT ............................................................................ 6
6. Застосування POSIT .......................................................................................... 7
7. Майбутнє POSIT та його потенційна заміна IEEE 754 .................................... 8
8. Реалізація .......................................................................................................... 8
9. Висновки ................................................................................................................................ 12
10. Джерела ................................................................................................................................. 13

**Вступ**

У сучасному світі чисельні обчислення відіграють важливу роль в різних сферах науки та інженерії. Точність та ефективність обчислень залежать від стандартів, що використовуються для подання та обробки чисел з рухомою комою. Стандарт IEEE 754, що домінує у цій області, був введений у 1985 році і з того часу став основою для чисел з рухомою комою у більшості обчислювальних систем.

Однак, з розвитком технологій та збільшенням потреб у точності обчислень, стандарт IEEE 754 поступово стикається з обмеженнями, які можуть впливати на продуктивність та надійність обчислювальних систем. Останнім часом виникає все більше досліджень, спрямованих на виявлення альтернативних підходів до подання чисел з рухомою комою.

Однією з таких альтернатив є числа типу POSIT, запропоновані вченим Джоном Густавсоном у 2017 році. Вони представляють собою новий формат чисел з рухомою комою, який обіцяє забезпечити вищу точність, кращу процесорну ефективність та енергоефективність порівняно зі стандартом IEEE 754.

У цій курсовій роботі ми розглянемо основні характеристики чисел типу POSIT, їх відмінності від чисел стандарту IEEE 754 та розглянемо потенціал заміни цього старого стандарту на новий. Ми розглянемо основні поняття, властивості та алгоритми операцій над числами типу POSIT, проаналізуємо їх переваги та недоліки, а також розглянемо можливі сценарії їх використання в різних додатках та обчислювальних системах

**Основи типу числа POSIT**

Тип числа POSIT є числовим форматом з плаваючою точкою, який був розроблений доктором Джоном Густавсоном в 2017 році. Він базується на ідеї універсальних чисел (унум), які мають фіксований розмір, але змінний діапазон та точність. Головна мета розробки POSIT була підвищення точності та ефективності обчислень з плаваючою точкою порівняно зі стандартом IEEE 754.

**Відмінності між POSIT та IEEE 754**

Основні відмінності між POSIT та IEEE 754 полягають в їх представленні чисел та способі кодування. Розглянемо ключові відмінності:

1. Кодування:

POSIT використовує ефективніший метод кодування чисел, що дозволяє отримувати вищу точність на тому ж розмірі числа, що і IEEE 754. Для кодування POSIT використовується схема зі змінною кількістю знаків для експоненти та мантиси, залежно від потреби, замість фіксованого розподілу, як у стандарті IEEE 754.

1. Точність:

POSIT надає вищу точність обчислень за рахунок більшого числа значущих бітів для мантиси та експоненти. Це дозволяє забезпечити кращу точність, ніж IEEE 754, навіть при меншому розмірі числа.

1. Діапазон:

POSIT має змінний діапазон, що дозволяє забезпечити ширший діапазон чисел, ніж IEEE 754, при тому ж розмірі числа. Таким чином, POSIT може представляти числа з більшим діапазоном та точністю, ніж IEEE 754.

1. Представлення нуля та нескінченності:

В POSIT єдиний нуль (інформативний нуль) замість двох нулів (додатний та від'ємний) у IEEE 754. Відсутність нескінченності та NaN (Not a Number) спрощує операції з числами та зменшує кількість можливих помилок.

**Принцип роботи типу числа POSIT**

Загальна структура POSIT включає обов'язковий знаковий біт, один або декілька бітів режиму, декілька опціональних бітів експоненти та декілька опціональних бітів **мантіси** (див. рис. 1). Знаковий біт дорівнює **0** для позитивних чисел і **1** для негативних. Кількість бітів режиму визначається динамічно залежно від спеціального кодування. Після знакового біта режим містить послідовність 0 або 1, яка закінчується протилежним бітом ( r̄ ) або на кінці числової форми. Аналогічно, кількість бітів для експоненти та мантіси визначається динамічно. POSIT число містить показник степеня та дріб у разі потреби.A screenshot of a computer

Description automatically generated with medium confidence

Рис.1: Загальний формат положення для скінченних ненульових значень кодів.

Нехай m - кількість однакових бітів у режимних бітах. Якщо перший біт дорівнює нулю, кількість нулів (m) відображає від'ємне значення (-m). В протилежному випадку, кількість одиниць зменшена на один (m-1) відображає позитивне значення (m-1). Біти режиму втілюють масштабний коефіцієнт useedk , де useed = 22es.

Експонента e розглядається як беззнакове ціле число для реалізації іншого масштабу 2*e*. На відміну від IEEE 754, POSIT не використовує зміщення для експоненти. Кожна експонента може мати задану кількість бітів ( es ). Залишкові біти після режиму та експоненти використовуються для мантіси (f). На зразок IEEE 754, мантіса включає прихований біт, який завжди дорівнює 1, оскільки позит не містить денормалізованих чисел. Загалом, n-бітне число (p) може представляти наступні числа.

Отже, універсальний формат POSIT складається з різних бітів, що динамічно відповідають різним числовим представленням. Знаковий біт вказує на знак числа, біти режиму допомагають визначити масштабний коефіцієнт, біти експоненти слугують для задання масштабу, а біти мантіси відповідають за частку числа. Ця структура дозволяє POSIT забезпечувати гнучке кодування чисел з різною точністю та масштабом, забезпечуючи ефективність обчислень та зберігання інформації.

Текст, лист

Опис генерується автоматично

Наприклад, рис.3 представляє 477/134217728 ≈ 3,55393 × 10-6, де es=3.

Життєпис

Опис генерується автоматично

Рис. 3: Приклад POSIT числа та його десяткового значення

**Переваги та недоліки POSIT**

Переваги:

1. Унікальне представлення значень: У POSIT, результати арифметичних операцій є однаковими на різних апаратних системах, а також дорівнюють одне одному, якщо використовують однакові бітові шаблони.
2. Відсутність поступового недостатнього заповнення: POSIT виключає проблему недостатнього заповнення завдяки підтримці змінної точності.
3. Збереження алгебраїчних правил у різних форматах: POSIT забезпечує асоціативність додавання та гарантує однакові результати при обчисленні значень на основі декількох форматів різного розміру.
4. Спрощення обробки виключень: Відсутність NaN у форматі POSIT та єдині представлення для 0 та ∞ спрощують обчислення.

Недоліки:

1. Декодування полів: Послідовне декодування полів POSIT через використання динамічних діапазонів може бути менш ефективним, ніж паралельне декодування у числах з плаваючою точкою.
2. Новизна системи: На відміну від системи чисел з плаваючою точкою, POSIT є новою числовою системою, яка ще має бути досліджена та реалізована у чіпах.

**Застосування POSIT**

Завдяки своїм перевагам, тип числа POSIT може бути використаний у різних сферах, таких як:

* Обчислювальна техніка, де потрібна висока точність обчислень та оптимізація ресурсів.
* Візуалізація даних, де потрібна висока точність представлення чисел та широкий діапазон значень.
* Машинне навчання та штучний інтелект, де важлива точність обчислень для оптимізації моделей та алгоритмів.
* Наукові дослідження, де потрібна висока точність та широкий діапазон чисел для моделювання та аналізу різноманітних процесів.
* Робототехніка та вбудовані системи, де важлива оптимізація ресурсів та швидкість обчислень.

**Майбутнє POSIT та його потенційна заміна IEEE 754**

Хоча тип числа POSIT має значні переваги порівняно зі стандартом IEEE 754, його широке впровадження та заміна IEEE 754 залежить від декількох факторів:

* Розвиток апаратного та програмного забезпечення, яке підтримує POSIT числа.
* Створення стандартів для використання POSIT чисел у різних галузях та інтеграція з існуючими системами.
* Вирішення можливих проблем зі сумісністю та надійністю операцій з POSIT числами.
* Освітні та наукові програми для підвищення обізнаності про POSIT та його можливості серед науковців, інженерів та розробників.

**Реалізація**

Весь код знаходиться в папці src. Реалізацію Posit розділено на 5 файлів:

Скріншот комп'ютера

Опис автоматично створюється з низькою впевненістю

operations. cpp – файл з допоміжними функціями та алогоритмами математичних операцій (operations. h – хедер файл для нього)

posit. cpp – файл з реалізацією класу Posit (posit. h – хедер файл для нього)

posit\_types. h – файл з типами для легшого орієнтуванні в коді.

У моїй реалізації на С++ клас POSIT має 3 основні компоненти: знаковий біт, експоненту1 та мантісу. Під час математичних операцій експонента1 розкладається на 2 частини: експоненту2 та режім.

Код для тестів знаходиться у папці TESTS.

A picture containing text, font, screenshot, white

Description automatically generated

posit\_test.cpp – файл з тестами для класу Posit де перевіряється ввід та вивід, робота математичних операцій, робота математичних фунції (<><=>==), робота стандарту числа у циклі та підрахунок часу для циклу.

A picture containing text, font, screenshot

Description automatically generated

A picture containing font, text, graphics, white

Description automatically generatedЯк використовувати програму:

- Щоб створити новий екземпляр класу:

- Усі математичні операцію та вивід для класу Posit перегружені.

Як працює програма:

- У файлі posit\_types.h всі основні типи чисел, які використовуються у програмі.

- Реалізовано 3 методи класу для створення Posit числа. А саме для початкових чисел: int, float, double

A picture containing font, text, design

Description automatically generated

- При створенні екземпляру класу, програма дане число (int, float, double) розбиває на такі елементи:

A picture containing text, font, electric blue, screenshot

Description automatically generated

Де number – саме число, number\_size – довжина числа у бітовому представлені, exp\_size – довжина експоненти1 (яка є комбінацією експоненти2 та режіму)

- При математичних операціях, ці значення класу передаються у функцію unpack\_posit(), яка створює структуру:

A screen shot of a computer code

Description automatically generated with low confidence

Де neg – від’ємне чи додатнє число? , exp – експоненти1 (яка є комбінацією експоненти2 та режіму), frac – мантіса

 У цій структурі exp рахується за формулою де reg – режім, який рахується при розпакуванню number, number\_size та exp\_size .

- Після рахування новою структури unpack\_t для результуючого числа, ця структура передається у функцію pack\_posit(), щоб порахувати основний number. Та потім створюється нове число:



- Щоб подивитися числа у бітовій формі (а саме знак, режім, експонента, мантіса) треба використовувати ость такі функції:

A picture containing text, font

Description automatically generated

- Щоб вивести всю інформацію про число треба використовувати звичайний перегружений оператор std::cout .

- Щоб перевести Posit число у звичайне число з плаваючою комою(float, double), треба використовувати ці функції:

A picture containing font, text, line

Description automatically generated

**Приклад:**

1) Створимо Posit число з значенням 123.456.   
2) Покажемо як виглядає це число у бітовій формі. (знаковий біт - режім - експонента - мантіса).   
3) Продемонструємо роботу математичних операцій.

Зображення, що містить текст, шрифт, лінію, скріншот

Опис генерується автоматично

Зображення з текстом, шрифтом, знімком екрана, білим кольором

Опис генерується автоматично

Приклад іншої реалізації:

<https://github.com/eruffaldi/cppPosit> - також непогана реалізація стандарту числа Posit.

**Висновки**

Тип числа POSIT пропонує інноваційний підхід до представлення чисел з плаваючою точкою, який може підвищити точність та ефективність обчислень порівняно зі стандартом IEEE 754. Його переваги, такі як вища точність, ширший діапазон та ефективніше кодування, роблять його потенційною заміною для IEEE 754 у різних сферах застосування. Однак успішне впровадження POSIT залежить від розвитку підтримки в апаратному та програмному забезпеченні, створення стандартів та розв'язання можливих проблем зі сумісністю та надійністю. Освітні та наукові програми можуть сприяти підвищенню обізнаності про POSIT та його можливості, а також сприяти його широкому впровадженню та заміні IEEE 754 у майбутньому.

**Джерела**

* Густафсон, J. (2017). Перемога з плаваючою точкою у власній грі: позитивна арифметика. Кордони та інновації суперкомп'ютерів.
* <https://github.com/libcg/bfp>
* https://www.sigarch.org/posit-a-potential-replacement-for-ieee-754/
* https://en.wikipedia.org/wiki/Unum\_(number\_format)#Unum\_III