

*Цілочисельна
арифметика в
обмеженому числі
розрядів*

Довгі числа

- Відомо, що арифметичні дії, які виконуються комп'ютером в обмеженій кількості розрядів, не завжди дозволяють отримати точний результат.
- Для реалізації операцій з довгими числами необхідно розмістити довге число в пам'яті комп'ютера.
- Більш того, ми обмежені розміром (величиною) чисел, з якими можемо працювати.

Арифметичні дії

- Нам необхідно виконати арифметичні дії над дуже великими числами, наприклад
- $30! = 265252859812191058636308480000000$
- У таких випадках ми самі повинні потурбуватися про представлення чисел в машині і про точне виконання арифметичних операцій над ними. Числа, для представлення яких в стандартних комп'ютерних типах даних не вистачає кількості двійкових розрядів, називаються "довгими".

Реалізація арифметичних операцій

- Реалізація арифметичних операцій над такими "довгими" числами отримала назву «довгої арифметики». Організація роботи з "довгими" числами залежить від того, як ми представимо в комп'ютері ці числа.
- **"Довге" число можна записати, наприклад, за допомогою масиву десяткових цифр, кількість елементів в такому масиві дорівнює кількості значущих цифр в "довгому" числі.**

Арифметичні операції

- Якщо ми реалізовуватимемо арифметичні операції над цим числом, то розмір масиву має бути достатнім, аби розмістити в ньому і результат, наприклад, множення. Існують і інші представлення "довгих" чисел. Розглянемо одне з них.
- Представимо наше число
- $30! = 265252859812191058636308480000000$ у вигляді:
- $30! = 2 \cdot (10^4)^8 + 6525 \cdot (10^4)^7 + 2859 \cdot (10^4)^6 + 8121 \cdot (10^4)^5 + 9105 \cdot (10^4)^4 + 8636 \cdot (10^4)^3 + 3084 \cdot (10^4)^2 + 8000 \cdot (10^4)^1 + 0000 \cdot (10^4)^0$.

- Даний запис нашої хує на думку про масив, представлений в таблиці.

Номер елемента в масиві А	0	1	2	3	4	5	6	7	8	9
Значення	9	0	8000	3084	8636	9105	8121	2859	6525	2

Множення довгого числа на коротке

- Виконується множення "в стовпчик" на одноцифрове число. Дана операція може бути корисна для виконання простих операцій, таких як множення на 2 або використана при діленні довгого числа на довге.
- Різниця між множенням на коротке число і множенням на довге полягає в тому, що при множенні на коротке число нам не потрібно множити розряди довгого числа на відповідні розряди короткого числа.

Ми можемо вважати, що наше "довге" число представлене в 10000—10 системі числення (десятитисячно-десяткова система числення, а "цифрами" числа є чотиризначні числа.

- **Наприклад. Ввести "довге" число.**
 - Рішення задачі почнемо з опису даних.
- `Const MaxDig=1000;`
{Максимальна кількість цифр — чотиризначних!}
- `Osn=10000;`
- *{Основа нашої системи числення, в елементах масиву зберігаємо чотиризначні числа}*
- `Type Tlong=Array[0..MaxDig] Of Integer;`

Алгоритм введення "довгого" числа з файлу розглянемо на конкретному прикладі.

- Нехай в файлі записано число 23851674 і основою системи (O_{sn}) є 1000 (зберігаємо по три цифри в елементі таблиці A).
- Зміна значень елементів таблиці A в процесі введення (посимвольного в змінну ch) відображено в таблиці

A[0]	A[1]	A[2]	A[3]	ch	Примітка
0	0	0	0	2	Початковий стан
1	2	0	0	3	1-й крок
1	23	0	0	8	2-й крок
1	238	0	0	5	3-й крок
2	385	2	0	1	4-й крок: старша цифра елемента A [1] перейшла в поки "порожній" елемент A[2]
2	851	23	0	6	5-й крок
2	516	238	0	7	6-й крок
3	167	385	2	4	7-й крок
3	674	851	23	-	Кінцевий стан

- Якщо виводите масив, де в кожній комірці знаходиться по декілька цифр - можливі проблеми з виведенням ведучих нулів.
- Якщо нам потрібно вивести, наприклад, число 12030002, і в кожній комірці знаходяться по 4 цифри, то ми отримуємо масив з двох комірок, $T[1]=2$, та $T[2]=1203$.
- Під час виведення за допомогою функції, наведеної вище, ми отримаємо 12032. Ми втрачаємо три розряди, значення яких дорівнюють 0.

- В такому випадку необхідно застосувати форматований вивід. Для цього скористаємось двома флагами форматowanego виведення:

-

Додавання цілих довгих чисел. Процес додавання довгих чисел – це реалізація додавання чисел "в стовпчик". Перше число – масив T1, друге число – масив T2, результат – T3.

- **Наприклад:**

$$\begin{array}{r} ^1^1^1 \\ 67897 \text{ } T_1 \\ + \\ 328 \text{ } T_2 \\ \hline 68225 \text{ } T_3 \end{array}$$

Алгоритм процедури

- Алгоритм процедури додавання можна пояснити на простому прикладі.
- Нехай **A = 870613029451**,
- **B = 3475912100517461**

i	A[i]	B[i]	C[1]	C[2]	C[3]	C[4]
1	9451	7461	6912	1	0	0
2	1302	51	6912	1354	0	
3	8706	9121	6912	1354	7827	1
4	0	3475	6912	1354	7827	3476

Порівняння довгих чисел.

- Функція порівняння довгих чисел повертає 0 – коли числа рівні, 1 - коли перше число більше за друге і -1 - коли друге число більше першого.
- **Під час реалізації спершу порівнюється кількість цифр.**
- Якщо кількість цифр різна, то у більшого числа більше кількість цифр. Якщо ж кількість цифр однакова, то порівнюють всі цифри по порядку, починаючи зі старшого розряду.
- Даний алгоритм буде працювати правильно і у випадку коли в одній комірці пам'яті масиву будемо зберігати декілька цифр.