

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний аерокосмічний університет  
«Харківський авіаційний інститут»

Кафедра систем управління літальними апаратами

## ПОЗИЦІЙНІ СИСТЕМИ ЧИСЛЕННЯ

Національний аерокосмічний університет  
«Харківський авіаційний інститут»

Кафедра систем управління літальними апаратами

## ПОЗИЦІЙНІ СИСТЕМИ ЧИСЛЕННЯ

Розрахунково-графічна робота з дисципліни «Алгоритмізація і  
програмування»

ХАІ.301.174. 319.12 РГР

Виконав студент гр. 319

(№ групи)

\_\_\_\_\_ В. Є. Новиков  
(Підпис, дата) (П.І.Б.)

Перевірів к.т.н., доцент \_\_\_\_\_  
(Науковий ступінь, вчене звання)

\_\_\_\_\_ О. В. Гавриленко  
(Підпис, дата) (П.І.Б.)

## завдання

Дослідити шляхом власних обчислень, розробити і реалізувати алгоритми роботи з числами в різних позиційних системах числення:

- 1) Перетворити десяткові числа 126 в двійкову систему числення, описати покроково процес перетворень. Виконати перевірку, виконавши зворотне перетворення в десяткову систему.
- 2) Перетворити десяткові числа 1638 в шістнадцяткову систему числення, описати покроково процес перетворень. Виконати перевірку шляхом зворотного перетворення в десяткову і двійкову систему.
- 3) Розробити діаграму активності алгоритму перетворення числа з десяткової системи числення в 9-річну за алфавітом (0 1 2 3 4 5 6 7 F). \*Реалізувати алгоритм у вигляді строкової функції `DecTo_N_ (D)` з вхідним цілочисельним параметром на мові C ++.
- 4) Для двох чисел 126 та 1638 провести операцію множення у двійковій системі числення. Виконати перевірку шляхом перетворення результатів в десяткову систему.
- 5) Зробити висновки.

## ЗМІСТ

Вступ	4
1 Перетворення чисел в двійкову систему числення	5
1.1 Перетворення трирозрядного десяткового числа	5
1.2 Перетворення чотирирозрядного десяткового числа	5
1.3 Перевірка результатів	6
2 Перетворення чисел в шістнадцяткову систему числення	7
2.1 Перетворення трирозрядного десяткового числа	7
2.2 Перетворення чотирирозрядного десяткового числа	7
2.3 Перевірка результатів	8
3 Перетворення чисел в 9-річну систему числення	9
4 Двійкова арифметика	10
Висновки	11
Додаток А	12
Додаток Б	13

## Вступ

Всі позиційні системи числення "однакові", а саме, у всіх них арифметичні операції виконуються за одним і тим же правилам:

- справедливі одні й ті ж закони арифметики:
  - Комутативний
  - Асоціативний
  - Дистрибутивний
- справедливі правила додавання, віднімання, множення і ділення стовпчиком;
- Правила виконання арифметичних операцій спираються на таблиці додавання і множення.

# 1 Перетворення чисел в двійкову систему числення

## 1.1 Перетворення три-розрядного десяткового числа

Покроковий опис перетворення наведено у табл.1.1.

Таблиця 1.1 – Перетворення три-розрядного десяткового числа у двійкове

<b>X</b>	<b>X/2</b>	<b>X%2</b>
126	63	0
63	31	1
31	15	1
15	7	1
7	3	1
3	1	1
1	0	1
Результат		$126_{10} = 111110_2$

## 1.2 Перетворення чотири-розрядного десяткового числа

Покроковий опис перетворення наведено у табл.1.2.

Таблиця 1.2 – Перетворення чотири-розрядного десяткового числа у двійкове

<b>X</b>	<b>X/2</b>	<b>X%2</b>
1638	819	0
819	409	1
409	204	1
204	102	0
102	51	0
51	25	1
25	12	1
12	6	0
6	3	0
3	1	1
1	0	1
Результат		$1638_{10} = 11001100110_2$

### 1.3 Перевірка результатів

Перетворення з двійкової системи в десяткову:

$$1111110_2 = 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 64 + 32 + 16 + 8 + 4 + 2 + 0 = 126_{10}$$

$$11001100110_2 = 1 \times 2^{10} + 1 \times 2^9 + 0 \times 2^8 + 0 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 1 \times 1024 + 1 \times 512 + 0 \times 256 + 0 \times 128 + 1 \times 64 + 1 \times 32 + 0 \times 16 + 0 \times 8 + 1 \times 4 + 1 \times 2 + 0 \times 1 = 1024 + 512 + 0 + 0 + 64 + 32 + 0 + 0 + 4 + 2 + 0 = 1638_{10}$$

## 2 Перетворення чисел в шістнадцяткову систему числення

### 2.1 Перетворення три-розрядного десяткового числа

Покроковий опис перетворення наведено у табл.2.1.

Таблиця 2.1 – Перетворення десяткового числа у шістнадцяткове

X	X /16	X %16
126	7	14(E)
7	0	7
Результат		$126_{10} = 7E_{16}$

### 2.2 Перетворення чотири-розрядного десяткового числа

Покроковий опис перетворення наведено у табл.2.2.

Таблиця 2.2 – Перетворення десяткового числа у шістнадцяткове

X	X /16	X %16
1638	102	6
102	6	6
6	0	6
Результат		$1638_{10} = 666_{16}$

## 2.3 Перевірка результатів

Перетворення з шістнадцяткової системи у десяткову:

$$7E_{16} = 7 \times 16^1 + 14 \times 16^0 = 112 + 14 = 126_{10},$$

$$666_{16} = 6 \times 16^2 + 6 \times 16^1 + 6 \times 16^0 = 1536 + 96 + 6 = 1638_{10}$$

Перетворення в двійкову:

$$7 = 0111, E (14) = 1110, 7E_{16} = 01111110_2 = 1111110_2 \text{ (співпадає з пп.1.1)}$$

$$6 = 0110, 6 = 0110, 6 = 0110, 666_{16} = 011001100110_2 = 11001100110_2 \text{ (співпадає з пп.1.2)}$$



### 3 Перетворення чисел в 9-річну систему числення

Лістинг код на C++ представлено в додатку А (стор.12).

Діаграму активності представлено в додатку Б, рис. Б.1(стор.13).



## Висновки

*У результаті виконаної роботи було досліджено процеси переведення чисел між десятковою, двійковою та шістнадцятковою системами числення, виконано перевірки відповідності результатів, розглянуто множення двійкових чисел із графічним представленням операцій, створено діаграму активності алгоритму переведення десяткового числа в 9-річну систему числення з алфавітом (0 1 2 3 4 5 6 7 F) та реалізовано відповідний код мовою C++, що загалом дало змогу закріпити знання про різні системи числення, способи перетворення між ними, а також практично застосувати ці знання у вигляді програмної реалізації.*

## Додаток А

```

#include <iostream>
#include <string>
#include <algorithm> // Для reverse()

using namespace std;

// Функція для переведення десяткового числа в 9-річну систему з алфавітом
(0,1,2,3,4,5,6,7,F)
string DecTo_N_(int D) {
    // Обробка випадку, коли вхідне число дорівнює 0
    if (D == 0) {
        return "0"; // Будь-яка система числення: 0 залишається 0
    }
    string result; // Тут зберігатимемо результат

    // Головний цикл перетворення:
    while (D > 0) {
        int remainder = D % 9; // Знаходимо остачу від ділення на 9

        // Обробка особливого випадку: якщо остача 8, замінюємо її на 'F'
        if (remainder == 8) {
            result += 'F';
        }
        // Інакше додаємо цифру як символ
        else {
            result += to_string(remainder); // Конвертуємо цифру в рядок
        }

        D /= 9; // Ділимо число на 9 для наступної ітерації
    }

    // Реверсуємо рядок, тому що цифри отримувались у зворотньому порядку
    reverse(result.begin(), result.end());

    return result; // Повертаємо готовий результат
}

int main() {
    int number; // Змінна для зберігання вхідного числа

    // Запитуємо користувача ввести число
    cout << "Введіть десяткове число: ";
    cin >> number;

    // Конвертуємо число та виводимо результат
    string converted = DecTo_N_(number);
    cout << "Результат у 9-річній системі: " << converted << endl;

    return 0; // Кінець програми
}

```

## Додаток Б

На рис.Б.1 показано діаграму активності.

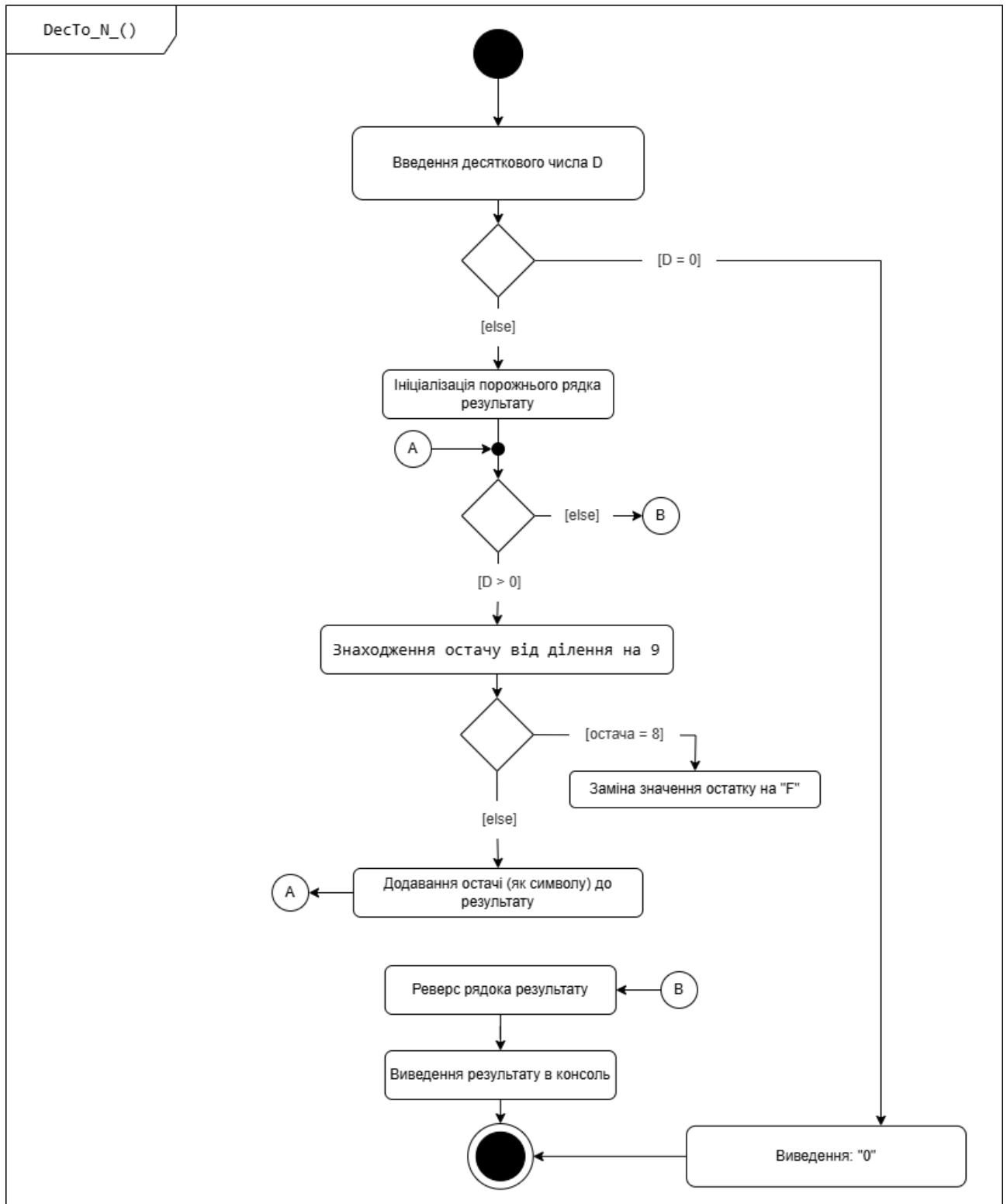


Рисунок Б.1 — діаграма активності.