

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра систем управління літальних апаратів

ПОЗИЦІЙНІ СИСТЕМИ ЧИСЛЕННЯ

Пояснювальна записка до розрахунково-графічної роботи
з дисципліни «Алгоритмізація та програмування»

XAI.301.174.312.8

Виконав студент гр. 312

_____ Діхтяренко М.Г.
(підпис, дата) (П.І.Б.)

Перевірив

_____ к.т.н., доц. Олена ГАВРИЛЕНКО
(підпис, дата) (П.І.Б.)

2024

Завдання

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

1. Перетворити десяткові числа в двійкову систему числення, описати покроково процес перетворень. Виконати перевірку, виконавши зворотне перетворення в десяткову систему.
2. Перетворити десяткові числа 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. *Реалізувати алгоритм у вигляді строкової функції `DecTo_N_ (D)` з вхідним цілочисельним параметром на мові C ++.
4. Для двох чисел провести операцію у двійковій системі числення. Виконати перевірку шляхом перетворення результатів в десяткову систему.
5. Зробити висновки.

Вступ

Системи числення:

- **Десяткова система** (також називається десятковою або знаковою) використовує 10 символів (0-9). Кожна позиція числа відображає степінь числа 10.
- **Двійкова система** використовує 2 символи (0 та 1). Кожна позиція числа відображає степінь числа 2.
- **Шістнадцяткова система** використовує 16 символів (0-9 та A-F), що відповідають десятковим числам від 0 до 15. Кожна позиція числа відображає степінь числа 16.
- **Вісімкова система** використовує 8 символів (0-7). Кожна позиція числа відображає степінь числа 8.
- **Поштовх дуже невеликий, деякі комп'ютери використовують системи числення на основі 3 або 12 (триадичну та дванадцяткову системи), але вони не так поширені, як ті, що були перераховані вище.

Двійкова арифметика:

- В двійковій системі числення кожна цифра (біт) може бути або 0, або 1.
- Додавання виконується так само, як і в десятковій системі, але з розширеним правилом, що при перевищенні значення 1 додається до наступного розряду, а в разі необхідності створюється новий розряд.
- Віднімання виконується аналогічно додаванню, але із зворотнім порядком дій.
- Множення і ділення в двійковій системі також можна виконати, використовуючи аналогічні правила, що і в десятковій системі, але з урахуванням того, що кожне число представлене у двійковій формі.

Лістинг коду наведено в дод. А (стор. 5).

Екран роботи програми наведено в дод. Б (стор. 9).

ВИСНОВКИ

В ході виконання роботи було розглянуто загальні відомості про системи числення і двійкову арифметику, вивчено перетворення чисел, отриманні знання відпрацьовані на практиці в написанні коду.

ДОДАТОК А

Лістинг коду програми

```
#include <iostream>
#include <string>
#include <sstream>

using namespace std;

void task1();
string Dec_To_Bin(int);

void task2();
void Dec_To_Six(int);

void task3();
void DecTo_N_(int);

void task4();

int main() {

    int choice = 0;

    while (choice != 5) {

        cout << "Choose task: "
             << "\n1.Task1"
             << "\n2.Task2"
             << "\n3.Task3"
             << "\n4.Task4"
             << "\n5.Exit" << endl;

        cin >> choice;

        switch (choice) {
            case 1: {
                task1();          // Task of converting the number 144 from
decimal to binary and hexadecimal
                break;
            }
            case 2: {
                task2();          // The task of converting the number 2160
from decimal to hexadecimal and binary
                break;
            }
            case 3: {
                task3();          // Conversion task from decimal to thirteen
```

```

        break;
    }
    case 4: {
        task4();          // Divide two numbers 144 and 2160 in binary
        break;
    }
    case 5: {
        cout << "Program is over!";
        break;
    }
    default: {
        cout << "Wrong one, try again";
    }
}

}

void task1() {
    int number = 144;

    cout << "In decimal = " << number << " to binary = " << Dec_To_Bin(number)
<< endl;
    cout << "In decimal = " << number << " to hexadecimal = ";
    Dec_To_Six(number);

    cout << "\n";
}

string Dec_To_Bin(int n) {
    string binary;

    while (n > 0) {

        binary.push_back('0' + n % 2);
        n /= 2;
    }

    reverse(binary.begin(), binary.end());

    return binary;
}

void task2() {
    int number = 2160;

    cout << "In decimal = " << number << " to binary = " <<
    Dec_To_Bin(number) << endl;
    cout << "In decimal = " << number << " to hexadecimal = ";
    Dec_To_Six(number);

    cout << "\n";
}

```

```

}
void Dec_To_Six(int n) {
    string binary;

    while (n > 0) {

        if (n % 16 < 10) {
            binary.push_back('0' + n % 16);
            n /= 16;
        }
        else if (n % 16 == 10) {
            binary.push_back('A' + n % 16 - 10);
            n /= 16;
        }

        else if (n % 16 == 11) {
            binary.push_back('B' + n % 16 - 11);
            n /= 16;
        }
        else if (n % 16 == 12) {
            binary.push_back('C' + n % 16 - 12);
            n /= 16;
        }
        else if (n % 16 == 13) {
            binary.push_back('D' + n % 16 - 13);
            n /= 16;
        }
        else if (n % 16 == 14) {
            binary.push_back('E' + n % 16 - 14);
            n /= 16;
        }
        else if (n % 16 == 15) {
            binary.push_back('F' + n % 16 - 15);
            n /= 16;
        }

    }

    reverse(binary.begin(), binary.end());

    cout << binary;
}

void task3() {
    int in;
    cout << "Enter a number: ";
    cin >> in;
    cout << "Changed number into binary: ";

    DecTo_N_(in);
}

```

```

        cout << endl;
    }

    void DecTo_N_(int n) {
        string binary;

        while (n > 0) {

            if (n % 13 < 10) {
                binary.push_back('0' + n % 13);
                n /= 13;
            }
            else if (n % 13 == 10) {
                binary.push_back('K' + n % 13 - 10);
                n /= 13;
            }
            else if (n % 13 == 11) {
                binary.push_back('L' + n % 13 - 11);
                n /= 13;
            }
            else if (n % 13 == 12) {
                binary.push_back('M' + n % 13 - 12);
                n /= 13;
            }

        }

        reverse(binary.begin(), binary.end());
        cout << binary;
    }

    void task4() {
        int number1 = 2160;
        int number2 = 144;

        cout << "Divide of numbers: " << number1 << " and " << number2 << " to
        binary system is " << Dec_To_Bin(2160/144) << std::endl;
    }

```


ДОДАТОК Б
Скрін-шот вікна виконання програми

```
Choose task:
1.Task1
2.Task2
3.Task3
4.Task4
5.Exit
1
In decimal = 144 to binary = 10010000
In decimal = 144 to hexadecimal = 90
Choose task:
1.Task1
2.Task2
3.Task3
4.Task4
5.Exit
2
In decimal = 2160 to binary = 100001110000
In decimal = 2160 to hexadecimal = 870
Choose task:
1.Task1
2.Task2
3.Task3
4.Task4
5.Exit
3
Enter a number: 3345
Changed number into binary: 16K4
Choose task:
1.Task1
2.Task2
3.Task3
4.Task4
5.Exit
4
Divide of numbers: 2160 and 144 to binary system is 1111
Choose task:
1.Task1
2.Task2
3.Task3
4.Task4
5.Exit
```