

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «Обчислювальної техніки та програмування»

Звіт з лабораторної роботи №6

Тема: «Масиви»

Виконав:

уч. гр. КІТ-120в Олексієнко Микита

Харків 2020

Мета: Виконати завдання із категорії “На відмінно”:

3. Перетворити число (максимальне значення якого - 9999) в рядок.

Наприклад,

- 123 – “Сто двадцять три”,
- 4321 – “Чотири тисячі триста двадцять один”.

Номер завдання я здобув за формулою

$$N_t = ((N_j - 1) \% C) + 1,$$

де:

- N_j – номер студента у журналі групи;
- C – кількість варіантів у лабораторній роботі (для розраховуемого рівня);
- $\%$ – Ділення з остачею.

Мій номер студента 15, тоді: $(15-1)/6+1=3$ (розглядаємо завдання 3)

Виконання роботи:

1. Створюємо змінні x , в яку ми будемо писати наше 4-х значне число.
2. Створюємо 3-х вимірний масив s , з назвами відповідних частин 4-х значного числа.
3. Робимо цикл `for`, визначає цифри числа, розпочинаючи зі старшої та доповнює строки s відповідною текстовою назвою, яка залежна від самої цифри та її позиції у числі.
4. Є числа по типу одиннадцять та шістнадцять (наприклад). Щоб програма виявила їх, треба використати перевірку `if`.
Якщо число закінчується на 10-19, то замість расшифрування окремих цифр числа використовується назва усього 2-х значного числа, яке знаходиться у 5 підрозділа масиву s . Крім того, ми достроково закінчуємо цикл, оскільки нам не треба обраблювати останню цифру.
5. Частини назви я розділяю пробілом.
6. По закінченню формування усієї строки, ми доповнюємо її символом кінця строки.
7. Закінчуємо програму `return 0`.

Мій код виглядає наступним образом:

```
#include <stdio.h>

int main ()
{
    int x = 6819;
    char a[100];
    int p = 0;
    int w = 1000;
    int z;

    const char s[5][10][20] = {
        { "", "one", "two", "three", "four", "five", "six", "seven", "eight", "nine" },
        { "", "ten", "twenty", "thirty", "fourty", "fifty", "sixty", "seventy", "eighty", "ninety" },
        { "", "one hundred", "two hundred", "three hundred", "four hundred", "five hundred", "six hundred", "seven hundred", "eight hundred", "nine hundred" },
        { "", "one thousand", "two thousand", "three thousand", "four thousand", "five thousand", "six thousand", "seven thousand", "eight thousand", "nine thousand" },
        { "ten", "eleven", "twelve", "thirteen", "fourteen", "fifteen", "sixteen", "seventeen", "eighteen", "nineteen" }
    };

    for ( int i = 3; i >= 0; i--)
    {
        z = x / w % 10;

        if ( i == 1 && z == 1 )
        {
            i = 4;
            z = x % 10;
        }

        for ( int c = 0; s[i][z][c] != 0; c++)
        {
            a[p++] = s[i][z][c];
        }

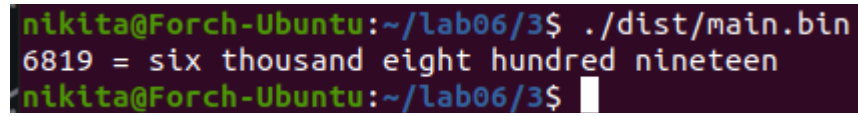
        a[p++] = ' ';
        w /= 10;

        if (i == 4) break;
    }

    a[--p] = '\0';
    printf("%i = %s\n", x , a);
    return 0;
}
```

Рисунок 1 — готовий код

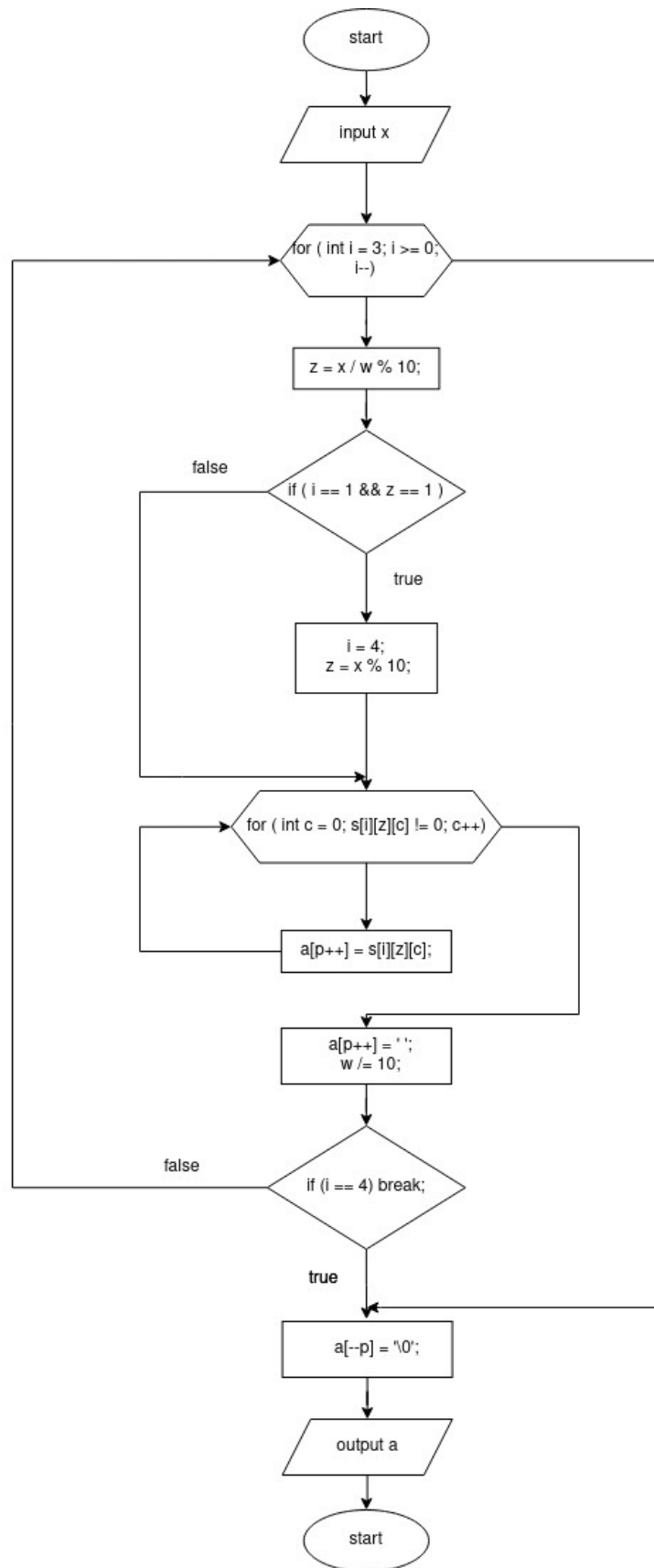
Я зробив виведення у термінал та отримав цей результат:



```
nikita@Forch-Ubuntu:~/lab06/3$ ./dist/main.bin
6819 = six thousand eight hundred nineteen
nikita@Forch-Ubuntu:~/lab06/3$
```

Рисунок 2 — Результат у терміналу

Також, для кращого сприйняття моєї програми я створим алгоритм:



Висновок.

В ході виконання даної Лабораторної роботи, я виконав 6 завдань, 1 з яких докладніше описав в цьому звіті. Це програма, яка перероблює числа в їх словесні версії.