

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «Обчислювальної техніки та програмування»

Звіт з лабораторної роботи №4

Тема: «Розробка лінійних програм, що розгалужуються»

Виконав:

уч. гр. КІТ-120в Олексієнко Микита

Харків 2020

**Мета: Виконати завдання із категорії “На відмінно”:**

3. Дано три числа  $k$ ,  $m$ ,  $n$ . Змінити значення змінних таким чином, щоб виконувалась умова  $k < m < n$ .

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 розраховуємого рівня);
- $\%$  – Ділення з остачею.

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

**Виконання роботи:**

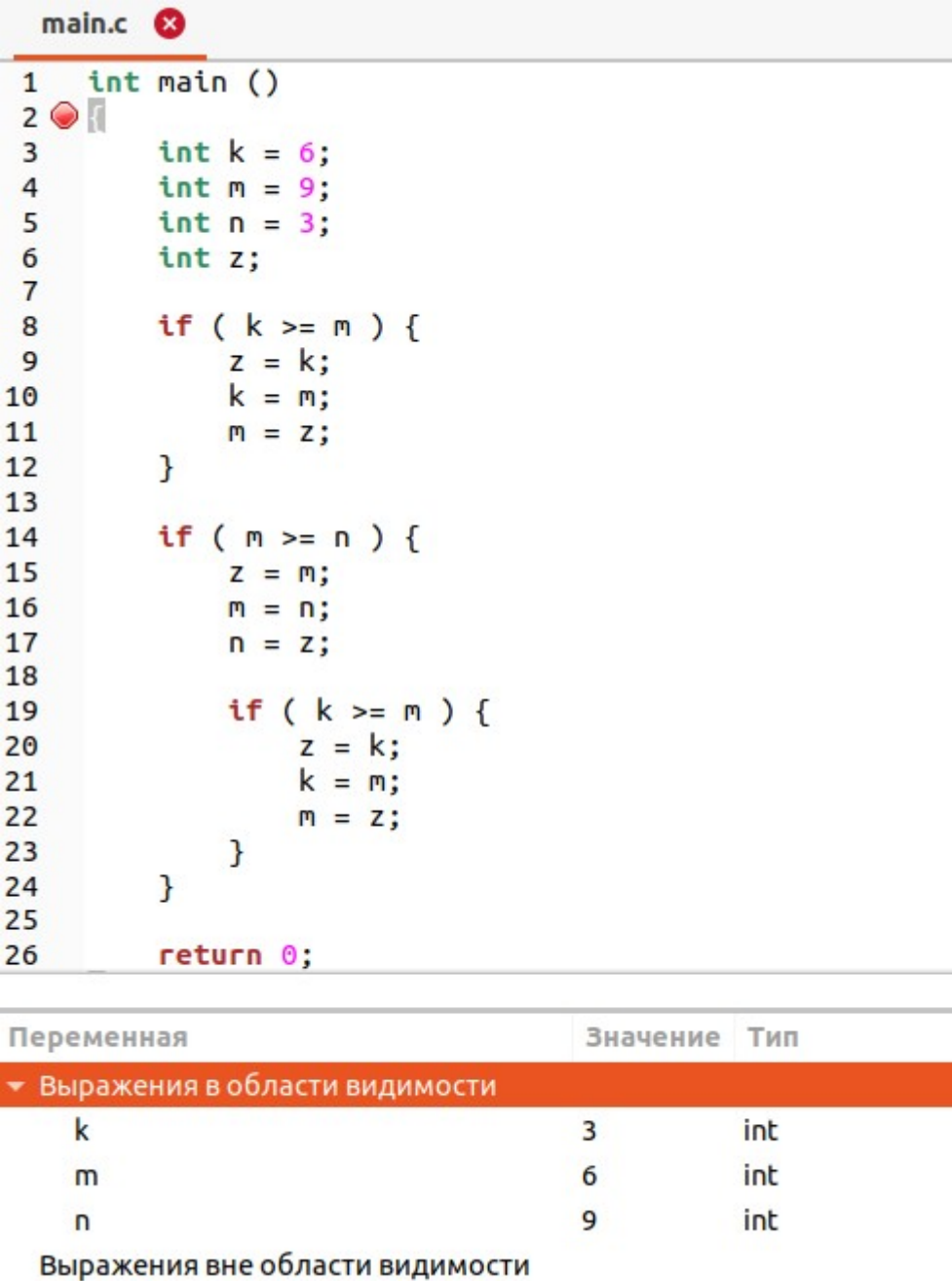
1. Ми створюємо змінні  $k$ ,  $m$  та  $n$ . Також нам потрібна змінна  $z$ , яка буде допоміжна.
  2. У своєму коді я буду використовувати метод бульбашки. Це простий спосіб розсортувати наші змінні у необхідному порядку.
  3. Вказуємо умову `if`, коли  $k > m$ . Потрібно також врахувати, що змінні можуть дорівнюватися, тож наша умова виглядає так:  $k \geq m$ .
  4. Перевіряємо змінні таким же чином:  $m \geq n$ .
  5. У останній дії ми оновили  $m$ , тому треба ще раз перевірити її з  $k$ .
- Я вирішив об'єднати `if` 4 дії з `if` 5 дії. Тому в моєму коді в `if` знаходиться ще один `if`.
6. Закінчуємо програму `return 0`.

Мій код виглядає наступним образом:

```
1  int main () |
2  {
3      int k = 6;
4      int m = 9;
5      int n = 3;
6      int z;
7
8      if ( k >= m ) {
9          z = k;
10         k = m;
11         m = z;
12     }
13
14     if ( m >= n ) {
15         z = m;
16         m = n;
17         n = z;
18
19         if ( k >= m ) {
20             z = k;
21             k = m;
22             m = z;
23         }
24     }
25
26     return 0;
27 }
```

Рисунок 1 — готовий код

Після дебагінгу, моя програма показала задовільний результат:

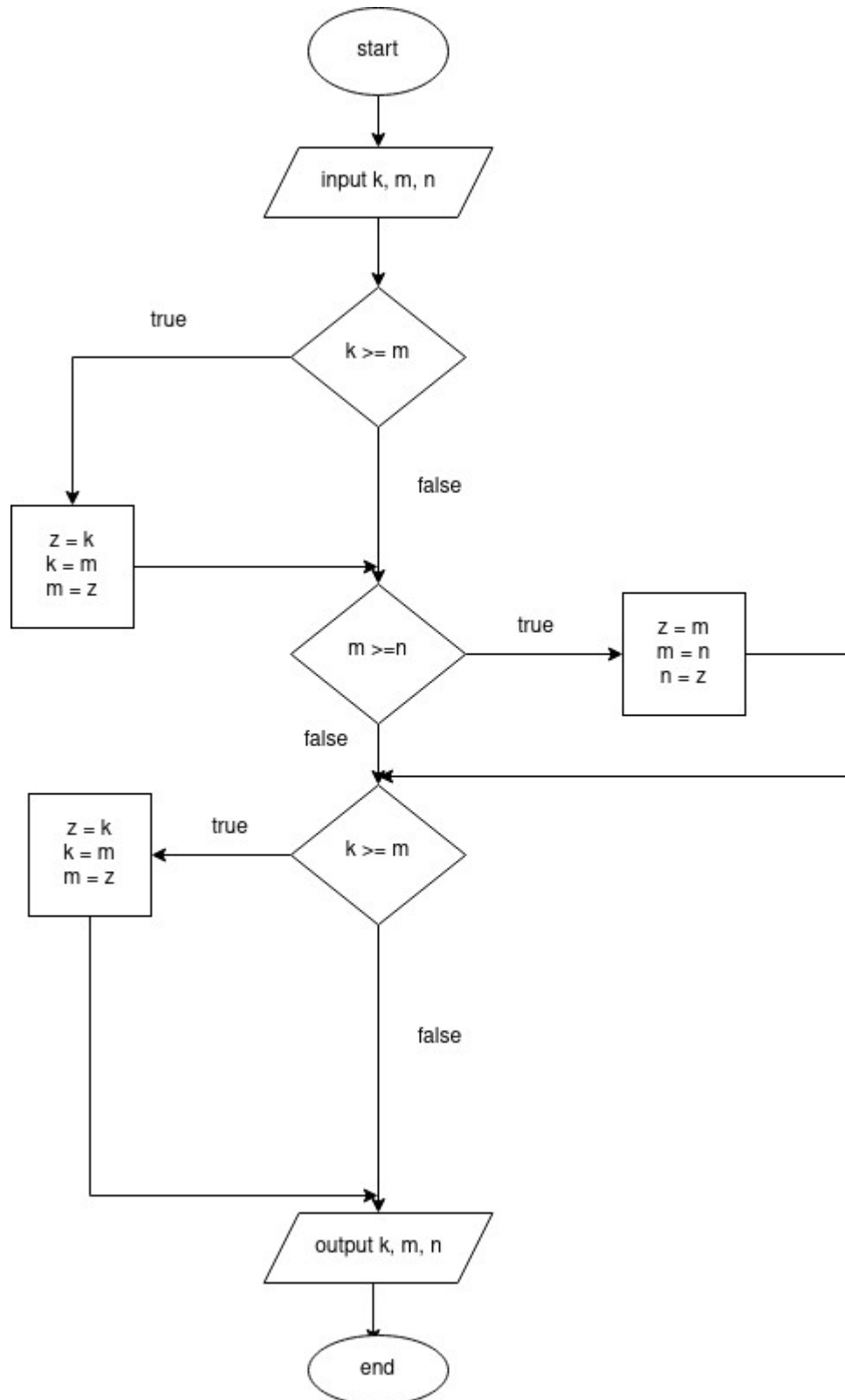


```
main.c x
1  int main ()
2  {
3      int k = 6;
4      int m = 9;
5      int n = 3;
6      int z;
7
8      if ( k >= m ) {
9          z = k;
10         k = m;
11         m = z;
12     }
13
14     if ( m >= n ) {
15         z = m;
16         m = n;
17         n = z;
18
19         if ( k >= m ) {
20             z = k;
21             k = m;
22             m = z;
23         }
24     }
25
26     return 0;
```

Переменная	Значение	Тип
▼ Выражения в области видимости		
k	3	int
m	6	int
n	9	int
Выражения вне области видимости		

Рисунок 2 — Код у Nemiver

Також, для кращого сприйняття моєї програми я створим алгоритм:



## Висновок.

В ході виконання даної Лабораторної роботи, я виконав 5 завдань, 1 з яких докладніше описав в цьому звіті. Це програма, яка порівнює числа та виставляє їх у порядку зростання.

## Відповіді на питання:

1. Оператор `if` працює наступним чином. Якщо вираз дає значення `true` (істина), тоді виконуються оператори відразу після слова `if`. В іншому випадку (значення виразу = `false`) виконуються оператори, які стоять після слова `else`.
2. Складений логічний вираз – це вираз, у якому використовуються логічні операції `&&` (and), `||` (or), `!` (not).
3. Оператор множинного вибору дозволяє робити вибір одного значення серед ряду перерахованих варіантів (оператор `switch`).
4. Значення виразу послідовно порівнюється з перерахованими варіантами із заданого списку. При виявленні збігу для однієї з умов порівняння виконується послідовність інструкцій, пов'язана з цією умовою.
5. Умовний (тернарний) оператор є єдиним тернарним оператором в мові C ++, який працює з 3-ма операндами. Через це його часто називають просто «тернарний оператор».  $(x = (\text{умова}) ? \text{значення1} : \text{значення2};)$
6. Вкладений умовний оператор `(?:)` застосовується для вибору до виконання одного з двох можливих операторів в залежності від деякої умови, при цьому один з операторів може бути відсутнім.
7. Бо будуть виконуватись інструкції, до тих пір, поки не зустрінеться інструкція `break`.
8. Так.
9. Так.