

Міністрество освіти і науки України  
Національний технічний університет  
«Харківський політехнічний інститут»  
Кафедра «Обчислювальна техніка та програмування»

### **ЗВІТ**

Про виконання лабораторної роботи № 3  
«Розробка лінійних програм»

Керівник: викладач  
Бульба С. С.

Виконавець: студент гр. КІТ-120в  
Стегній Б. В.

Харків 2020

# Лабораторна робота № 3. Розробка лінійних програм.

## 1 Вимоги

### 1.1 Розробник

- Стегній Богдан Володимирович;
- Студент групи КІТ-120в;
- 25-жов-2020.

### 1.2 Загальне завдання

**На оцінку “відмінно”.** Необхідно виконати усі завдання з даної категорії (проте звіт та відповідні зміни до системи контролю версіями виконуються лише за одним обраним студентом варіантом).

1. За заданим опором трьох резисторів  $r_1, r_2, r_3$ , які з'єднані паралельно, визначити загальний опір.
2. Визначити число, яке отримане виписуванням у зворотному порядку цифр заданого тризначного числа в десятковій системі числення.
3. Дано дійсне число  $a$ . Користуючись тільки операціями множення, отримати значення:  $a^4$  – за дві операції,  $a^6$  та  $a^8$  – за три операції,  $a^{10}$  – за чотири операції.
4. Дано 4-розрядне число у системі числення  $p$  (наприклад, 8). Визначити його еквівалент у десятковій системі числення.
5. Визначити, у скільки разів перша цифра 3х-значного числа більша, ніж остання. Результат «обрізати» до другого знака після коми. Наприклад,  $x = 123 \rightarrow y = 1/3 = 0.333333 = 0.330000$ .
6. Підрахувати суму чисел у заданому діапазоні. Наприклад, при вхідних даних 50 та 52 повинно бути  $50 + 51 + 52 = 153$ .

### 1.3 Індивідуальне завдання

За даною формулою обчислив варіант для написання звіту

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

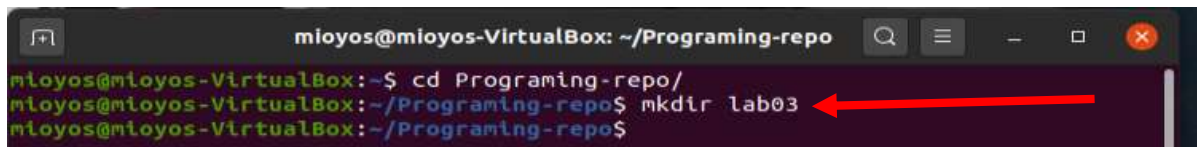
де:

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

3. Дано дійсне число  $a$ . Користуючись тільки операціями множення, отримати значення:  $a^4$  – за дві операції,  $a^6$  та  $a^8$  – за три операції,  $a^{10}$  – за чотири операції.

## 2 Виконання роботи

2.1 Створив директорію lab03 у репозиторії Programing-repo (Рис. 1, 2).



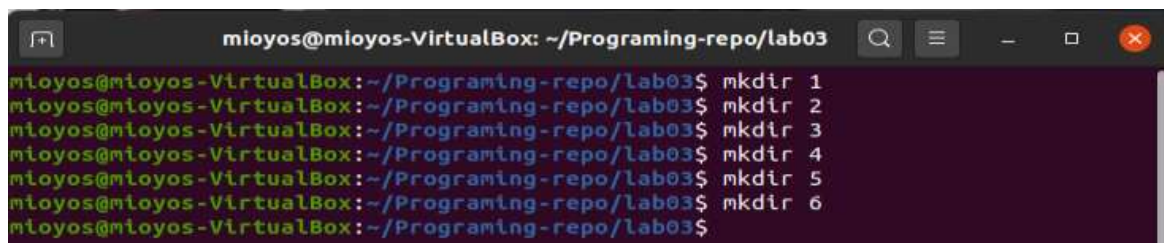
```
mloyos@mloyos-VirtualBox: ~/Programing-repo
mloyos@mloyos-VirtualBox:~$ cd Programing-repo/
mloyos@mloyos-VirtualBox:~/Programing-repo$ mkdir lab03
mloyos@mloyos-VirtualBox:~/Programing-repo$
```

(Рисунок 1)



(Рисунок 2)

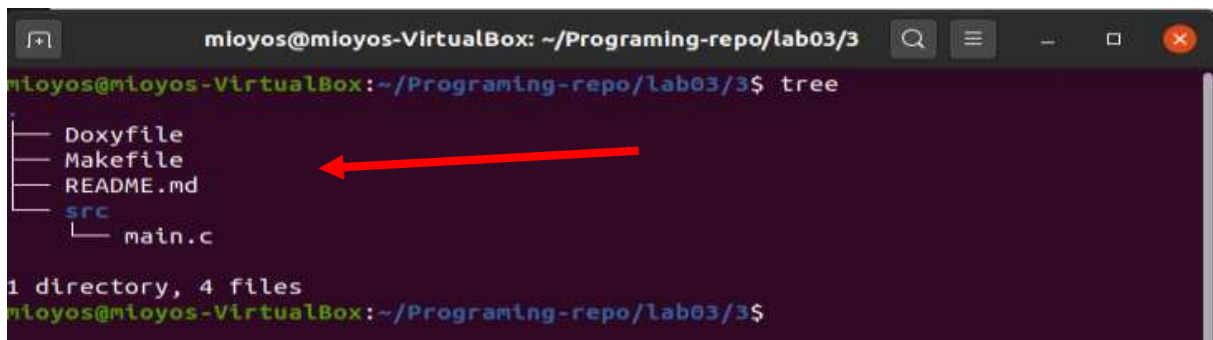
2.2 Створив директорії 1-6 за допомогою команди mkdir (Рис. 3).



```
mloyos@mloyos-VirtualBox: ~/Programing-repo/lab03
mloyos@mloyos-VirtualBox:~/Programing-repo/lab03$ mkdir 1
mloyos@mloyos-VirtualBox:~/Programing-repo/lab03$ mkdir 2
mloyos@mloyos-VirtualBox:~/Programing-repo/lab03$ mkdir 3
mloyos@mloyos-VirtualBox:~/Programing-repo/lab03$ mkdir 4
mloyos@mloyos-VirtualBox:~/Programing-repo/lab03$ mkdir 5
mloyos@mloyos-VirtualBox:~/Programing-repo/lab03$ mkdir 6
mloyos@mloyos-VirtualBox:~/Programing-repo/lab03$
```

(Рисунок 3)

2.3 Заніс потрібні файли до директорій для того, щоб можна було писати лінійні програми (Рис. 4, 5).



```
mloyos@mloyos-VirtualBox: ~/Programing-repo/lab03/3
mloyos@mloyos-VirtualBox:~/Programing-repo/lab03/3$ tree
.
├── Doxyfile
├── Makefile
├── README.md
└── src
    └── main.c

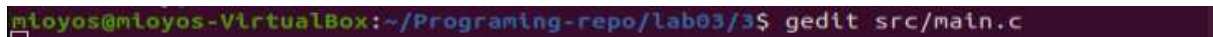
1 directory, 4 files
mloyos@mloyos-VirtualBox:~/Programing-repo/lab03/3$
```

(Рисунок 4)



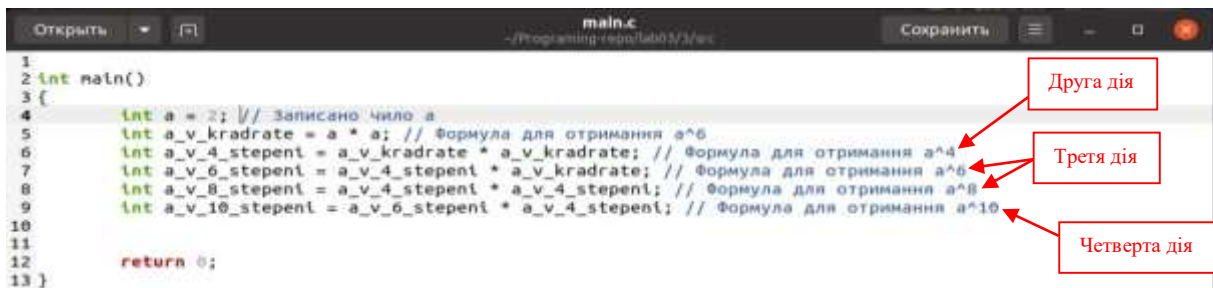
(Рисунок 5)

2.4 Відкрив вікно редагування коду за допомогою команди `gedit src/main.c` (Рис. 6).



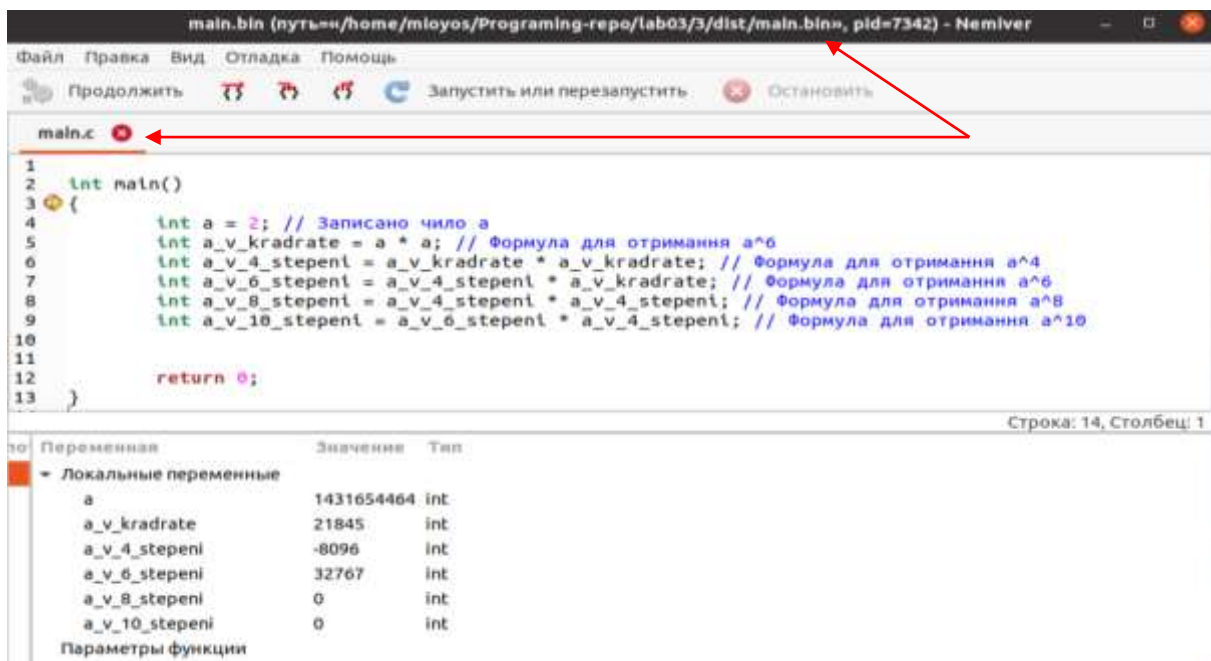
(Рисунок 6)

2.5 Додав усі потрібні змінні та шляхом їх множення піднесення до потрібного степеня за певну кількість операцій, яка вказана у лабораторному практимумі. Додав потрібні коментарі до коду (Рис. 7).



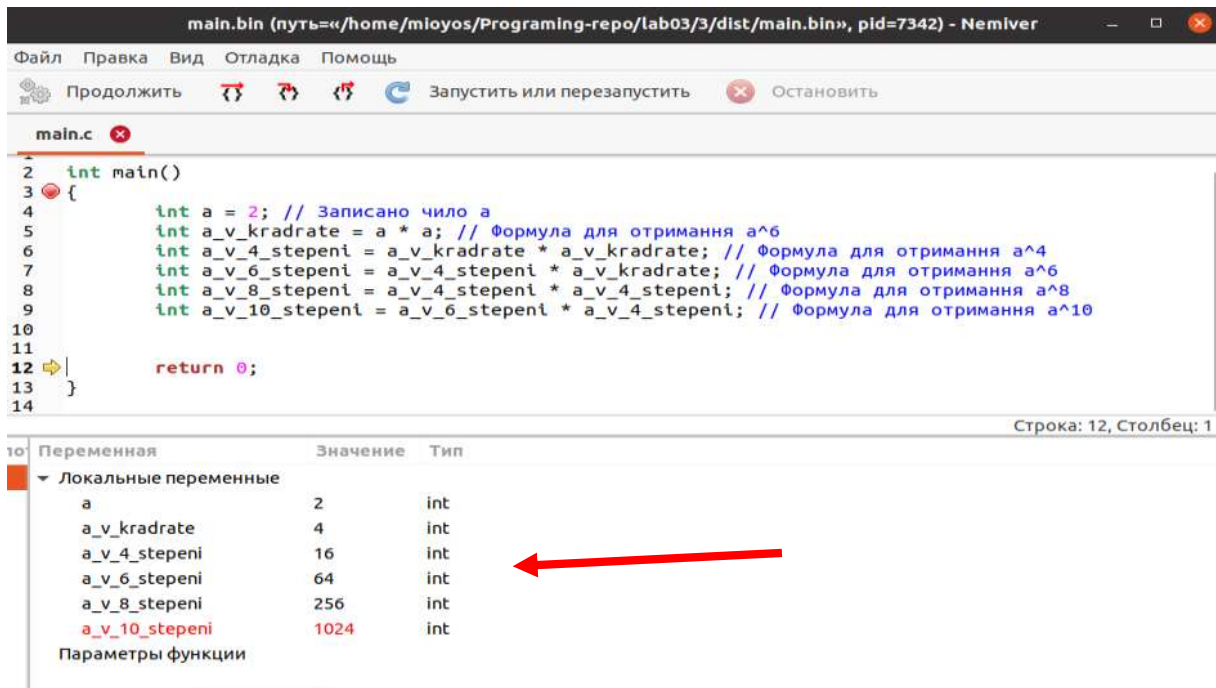
(Рисунок 7)

2.6 За допомогою команди `make clean prep compile` скомпілював проект та відкрив його у дебагері `nemiver` (`nemiver ./dist/main.bin`) (Рис. 8).



(Рисунок 8)

## 2.7 Перевірів працездатність програми (Рис. 9).



(Рисунок 9)

## 2.8 Усі зміни були зафіксовані (git commit -m "lab03") та завантажені до онлайн-репозиторію Programing-repo за допомогою команди git push (Рис. 10, 11).

```
mioyos@mioyos-VirtualBox:~/Programing-repo$ git commit -m "lab03"
[main db7ac00] lab03
36 files changed, 3729 insertions(+)
```

(Рисунок 10)

```
mioyos@mioyos-VirtualBox:~/Programing-repo$ git push
Username for 'https://github.com': mioyosss
Password for 'https://mioyosss@github.com':
Перечисление объектов: 36, готово.
Подсчет объектов: 100% (36/36), готово.
Сжатие объектов: 100% (28/28), готово.
Запись объектов: 100% (35/35), 9.06 KiB | 4.53 MiB/s, готово.
Total 35 (delta 6), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (6/6), completed with 1 local object.
To https://github.com/mioyosss/Programing-repo
4e48380..db7ac00 main -> main
```

(Рисунок 11)

## Висновки

При виконанні лабораторної роботи № 3 «Розробка лінійних програм», розробив 6 лінійних програм. Розробив програму, яка може підносити до степеню числа. Були розвинуті нові навички у роботі з середовищем Linux.



## Відповіді на контрольні запитання

1. Основними частинами типової структури програми на C++ є такі:
  - директиви препроцесорної обробки;
  - опис зовнішніх змінних (вихідних даних і результатів) та функцій;
  - функції програми;
  - головна функція — програми `main()`.
2. Лінійний алгоритм - це алгоритм, дії в якому виконуються послідовно, одна за одною.
3. `a++`, `++a`, `a=a+1`, `a +=1`.
4. Проект – це контейнер для усіх цих програм.
6. Треба завантажити компілятор (наприклад `gcc`) і виконати команду для компіляції (`gcc -o <назва_файла>`) (можна додати ключ `"-c"` якщо потрібна тільки компіляція або `"-g"` якщо треба додати інформацію відладника).
7. `git status` → `git commit` → `git push`.
8. За допомогою дебагера.
9. Запустити виконуючий файл (наприклад, `./main.bin`).
10. Консольні призначені більшою мірою для роботи в автоматизованому режимі з використанням скриптів та параметрів командного рядка. Віконні програми є більш дружніми до користувача, дозволяючи використовувати можливості графічного інтерфейсу користувача.
11. Змінні на відміну від констант можуть змінювати своє значення в процесі виконання програми.
12. Відмінність між префіксною та постфіксною формами операторів інкременту (`++`) та декременту (`--`) проявляється, коли ці оператори беруть участь в операції присвоювання.

Якщо вираз `++x` використовується в операторі присвоєння: `y = ++x`, то спочатку значення `x` збільшується на 1, а потім результуюче значення присвоюється змінній `y`.

Якщо виконати вираз `y = x++`, то спочатку змінній `y` присвоюється значення `x`, а потім значення `x` збільшується на 1.