

Лабораторна робота № 3

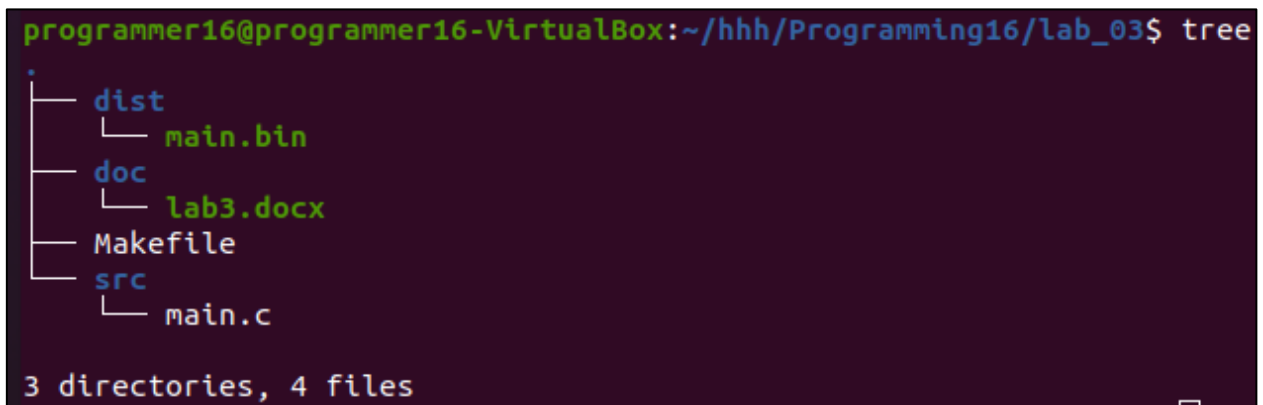
Тема: Розробка лінійних програм

Розробник: Студент 1-го курсу, Михневич Владислав Вікторович,
групи КІТ-120а.

Загальне завдання: За заданим опором трьох резисторів r_1 , r_2 , r_3 , які з'єднані паралельно, визначити загальний опір.

Індивідуальне завдання: За формулою визначити номер завдання для виконання звіту.

1. Створимо у папці свого репозиторія папку lab_03, в яку буду додавати нові файли (Рис. 1).



```
programmer16@programmer16-VirtualBox:~/hhh/Programming16/lab_03$ tree
.
├── dist
│   └── main.bin
├── doc
│   └── lab3.docx
├── Makefile
└── src
    └── main.c

3 directories, 4 files
```

Рисунок 1- Структура проекту.

2. Створимо тіло мови C (Рис. 2).

```
int main(){

return 0;
}
```

3. Задамо константи типу “float” (Рис. 2).

4. Задамо змінну “r”, яка в кінці буде дорівнювати нашому результату (Рис. 2).

5. Переробивши формулу опору, запишемо її до нашого результату змінної “ $r = (r_1 * r_2 * r_3) / (r_2 * r_3 + r_1 * r_3 + r_2 * r_1)$ ” (Рис. 2).

6. Запустимо програму та перевіримо за допомогою відлагодника та побачимо що все гаразд (Рис. 2).

```
1  #include <stdio.h>
2
3  int main(){
4  const float r1 = 2;
5  const float r2 = 3;
6  const float r3 = 4;
7  float r;
8  r = (r1 * r2 * r3) / (r2 * r3 + r1 * r3 + r2 * r1);
9
10
11 return 0;
12
```

ID по	Кадр	Функция	Переменная	Значение	Тип
1	0	main	Локальные переменные		
			r1	2	const float
			r2	3	const float
			r3	4	const float
			r	0	float

Рисунок 2 - Перевірка у відлагоднику.

— Точка входу знаходиться на третьому рядку на початку функції main, вона одна тому, що програма дуже проста і виконує одну головну дію (Рис. 3).

```
3  int main(){
4  const float r1 = 2;
5  const float r2 = 3;
```

Рисунок 3 - Точка входу.

8. Також відтворимо зміну стану на льоту змінивши значення «const float r1, r2, r3» на десяткове число, значення змінної “r” відобразилося правильно з цього доходимо висновку що все гаразд (Рис. 4).

```
3  int main()
4  {
5      const float r1 = 2.5;
6      const float r2 = 3.5;
7      const float r3 = 4.5;
8      float r;
9      r = (r1 * r2 * r3) / (r2 * r3 + r1 * r3 + r2 * r1);
10
11 return 0;
12 }
```

Local Variables	
Variable	Value
r1	2.5
r2	3.5
r3	4.5
r	1.10139859
Display Expressions	

Рисунок 4 - Запуск та зміна стану на льоту.

9. Після того як ми розробили та перевірили нашу програму можемо додати її до нашого репозиторію на Github. За допомогою послідовності команд:

```
git add .
```

```
git commit -m "Назва коміту"
```

```
git push
```

Висновок: На цій лабораторній роботі ми навчилися розробляти лінійні програми. Використовувати дії додавання(+), віднімання(-), множення(*), ділення(/), ділення з остачею(%). Перевіряти їх за допомогою відлагодника, змінювати стан та дивитися значення змінних за допомогою debug процесу.