

Лабораторна робота №6. Масиви

1 Вимоги

1.1 Розробник

- Кузнецов Микита Олександрович
- студент групи КІТ-320
- 09-dec-2020

1.2 Загальне завдання

Необхідно виконати усі завдання з даної категорії.

1.3 Індивідуальне завдання

Заповнити масив із заданої кількості елементів простими числами, що не повторюються. Розмір вихідного масиву задати наперед відомим значенням, що може бути більшим ніж результуюча кількість отриманих елементів.

Опис програми

1.4 Функціональне призначення

Програма призначена для заповнення масива із заданої кількості елементів простими числами, що декларовано в файлі `main.c`.

Результат заповнення зберігається у масиві `arrResult`

Демонстрація отриманих результатів передбачає покрокове виконання програми в режимі налагодження.

1.5 Опис логічної структури

За допомогою `#define ARR_SIZE` вказуємо розмір результуючого масива, через змінну `limit` вказуємо діапазон пошуку простих чисел.

Для отримання результату використовується функція `main`, що знаходиться в `main.c`

Структура проекту

```
. |— README.md
  |— src
    |— main.c
```

1.6 Важливі фрагменти програми

Початкові дані.

```
#define ARR_SIZE 30 // Размер массива
...
int limit = 100; // Диапазон поиска простых чисел
```

Пошук та запис простих чисел в масив

```
for (int i = 2; i < limit; i++) {
    for (int j = 2; j < i; j++) {
        if (i % j == 0) {
```

```

        check = 1; // Если check = 1 - число не простое
        break;
    } else {
        check = 0; // Если check = 0 - число простое
    }
}

if (check == 0) { // Число простое - записываем его в массив, если число не простое -
пропускаем
    arrResult[k] = i;
    k++;
}
}

```

2 Варіанти використання

Для демонстрації результатів використовується покрокове виконання програми в інтегрованому середовищі *Nemiver*. Нижче наводиться послідовність дій запуску програми у режимі відлагодження.

Крок 1 (див рис. 1). Знаходячись в основній процедурі, досліджуємо стан масива

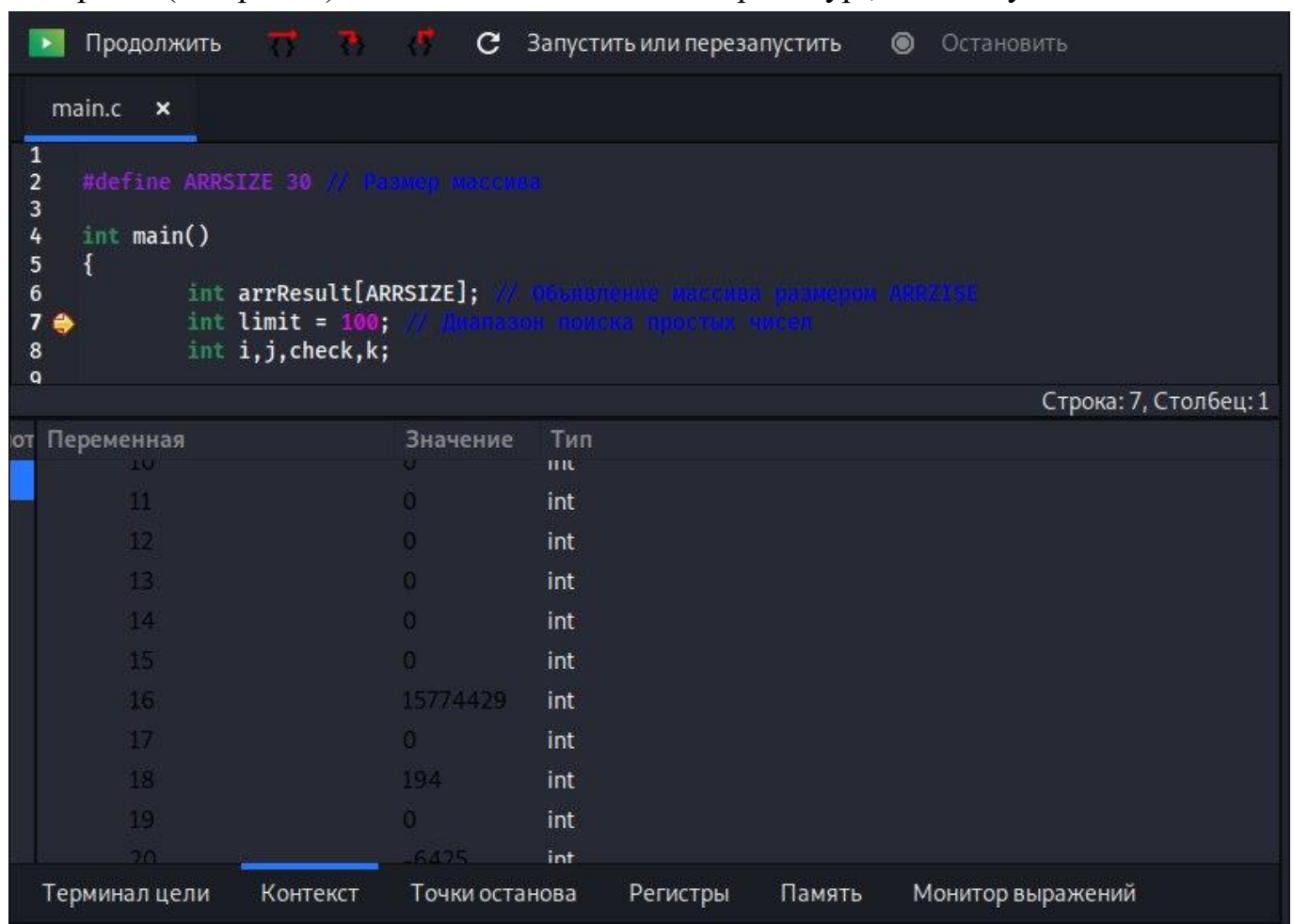


Рисунок 1 – вікно відлагодження в основній процедурі

Крок 2. Дослідження стану масива наприкінці виконання основної функції. Результат зображено на рис. 2, результат пошуку та запису простих чисел можна побачити в масиві `arrResult`

▶

Продолжить

⏮

⏪

⏩

↺

Запустить или перезапустить

⏸

Остановить

main.c x

21

arrResult[k] = 1;

22

k++;

23

24

}

25

}

26

27

return 0;

Строка: 27, Столбец: 1

от	Переменная	Значение	Тип
	Локальные переменные		
	arrResult	[30]	int [30]
	0	2	int
	1	3	int
	2	5	int
	3	7	int
	4	11	int
	5	13	int
	6	17	int
	7	19	int
	8	23	int
	9	29	int
	10	31	int
	11	37	int

Терминал цели

Контекст

Точки останова

Регистры

Память

Монитор выражений

Рисунок 2 – вікно відлагодження з результатом

3 Висновки

При виконанні даної лабораторної роботи було набуто практичного досвіду роботи з масивами та діями над ними.