

Лабораторна робота №8.2. Вступ до документації проекту

1. Вимоги

Розробник: Студент Теслюк Владислав Андрійович, КН-923г, вільний варіант, 08.12.23

2. Загальне завдання

Розробити повноцінний звіт для лабораторної роботи “Функції”, що присвячена функціям у двох форматів (+їх репрезентація у PDF форматі):

- Markdown
- .doc формат, згідно ДСТУ.

3. Опис програми

<https://github.com/Lordpluha/programming-Tesliuk/tree/main/lab%2008>

Призначення

Функція, що визначає, скільки серед заданої послідовності чисел таких пар, у котрих перше число менше наступного, використовуючи функцію з варіативною кількістю аргументів.

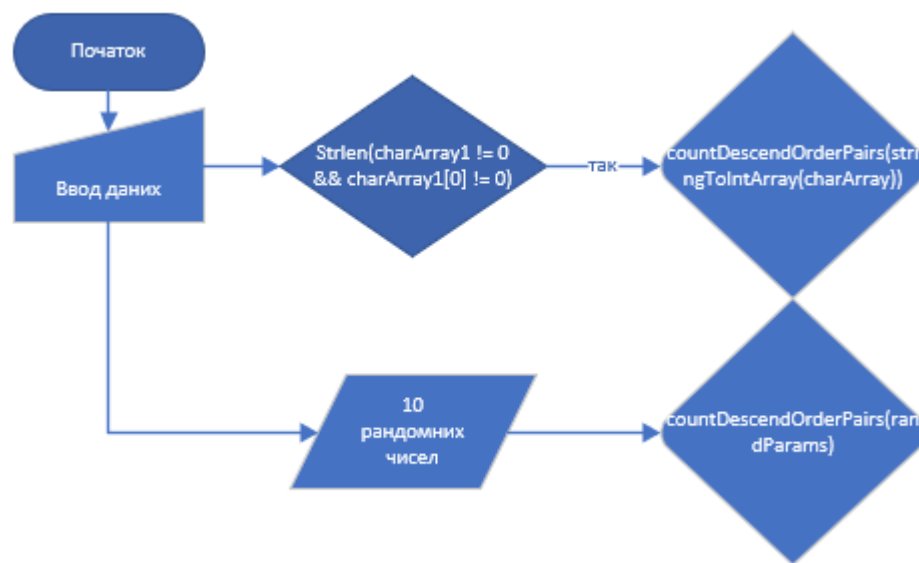
Наприклад, при вхідних даних { 3, 2, 4, 3, 1 }, результат повинен бути 3 (тобто наступні пари чисел: 3, 2, 4, 3, 3, 1)

Застосування

Формати вводу даних:

- {num1, num2, ...}
- num1, num2, ...
- num1, num2, ...
- num1 num2 ...

Опис логічної структури:



Опис констант/функцій:

memorySize – константа що відображає кількість чисел для вводу

```
struct arrayParams {  
    int lenght;  
    int* array;  
} - Структура для передачі даних через аргументи функцій
```

bool isDigit(char myChar) - Функція для перевірки чи є символ цифрою, приймає символ, повертає boolean

struct arrayParams stringToIntArray(char charArray[memorySize]) - Функція для парсинга чисел в строці в масив чисел без зайвих символів, приймає строку, повертає структуру типу arrayParams

int countDescendOrderPairs(struct arrayParams opts) - Функція для вирахування кількості пар зменшуючихся чисел, приймає структуру типу arrayParams, повертає число – результат програми

int main() – головна функція (entry point) у програмі

Структура директорії:

```
[parrot@parrot]--[~/programming-Tesliuk/lab 08]
$tree
.
├── dist
│   └── main.bin
├── doc
│   ├── lab 08.docx
│   ├── lab 08.md
│   └── lab 08.pdf
├── Doxyfile
├── Makefile
├── README.md
└── src
    └── main.c

3 directories, 8 files
```

Важливі фрагменти програми:

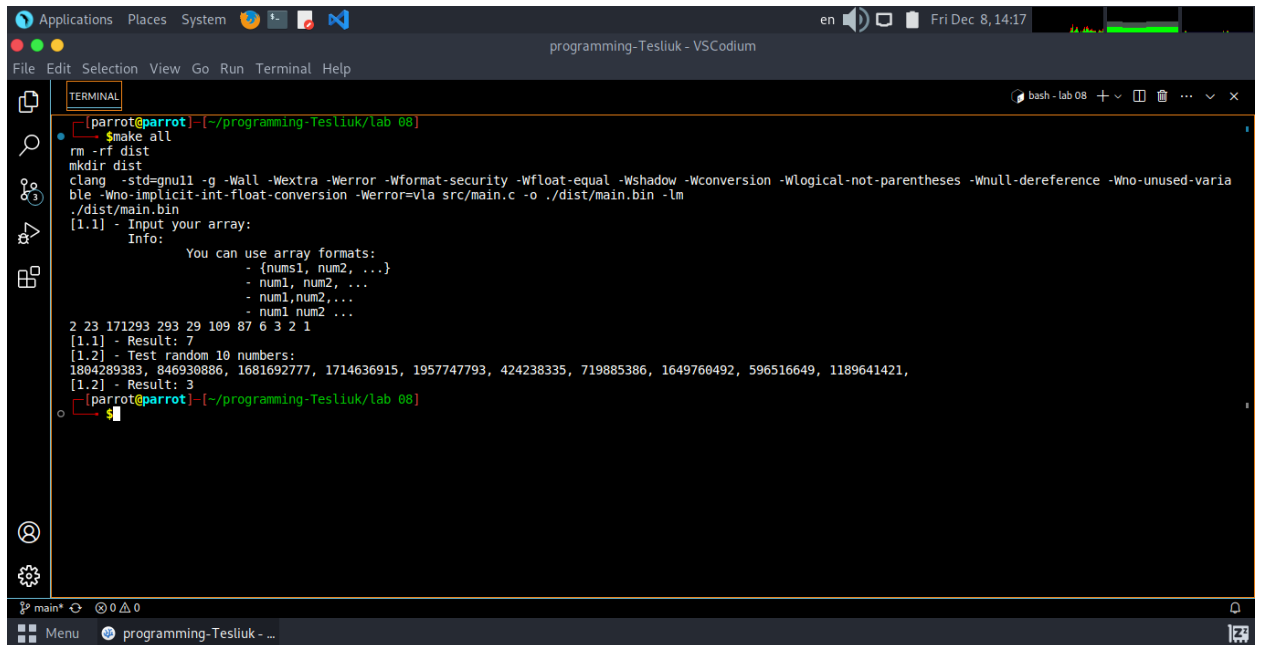
```
/// Функция для вычисления количества пар уменьшающихся чисел
int countDescendOrderPairs(struct arrayParams opts) {
    /// Деструктуризуем данные
    int lenght = opts.lenght;
    int *array = opts.array;

    /// Вычисляем количество пар
    int res = 0;
    for (int i = 0; i < (lenght - 1); i++) {
        if (array[i] > array[i+1]) {
            res++;
        }
    }

    /// Возвращаем результат
    return res;
}
```

struct arrayParams stringToIntArray(char charArray[memorySize]) - Функція для парсинга чисел в строці в масив чисел без зайвих символів, приймає строку, вертає структуру типу arrayParams

Варіанти використання:



```
parrot@parrot:~/programming-Tesliuk/lab 08$ make all
rm -rf dist
mkdir dist
clang -std=gnu11 -g -Wall -Wextra -Werror -Wformat-security -Wfloat-equal -Wshadow -Wconversion -Wlogical-not-parentheses -Wnull-dereference -Wno-unused-variable -Wno-implicit-int-float-conversion -Werror=vla src/main.c -o ./dist/main.bin -lm
./dist/main.bin
[1.1] - Input your array:
Info:      You can use array formats:
          - {nums1, num2, ...}
          - num1, num2, ...
          - num1,num2,...
          - num1 num2 ...
2 23 171293 293 29 109 87 6 3 2 1
[1.1] - Result: 7
[1.2] - Test random 10 numbers:
1804280383, 846930886, 1681692777, 1714636915, 1957747793, 424238335, 719885386, 1649760492, 596516649, 1189641421,
[1.2] - Result: 3
parrot@parrot:~/programming-Tesliuk/lab 08$
```

Висновки

Виконавши цю лабораторну роботу, я розробив повноцінні звіти для лабораторної роботи “Функції”, що присвячена функціям у двох форматах(**docx, md**) (+їх репрезентація у PDF форматі)