

Кожевніков Ілля КН-923в  
Лабораторна робота №8,2  
Вступ до документації проекту

Структура проекту:

```
kozhevka@AbobaLaptop:~/programming-kozhevnikov/lab08.2$ tree
.
├── Makefile
├── README.md
├── build
│   └── main.bin
└── scripts
    └── main.c
```

Додана функція check:

```
CC = clang
C_OPTS = $(MAC_OPTS) -std=gnu11 -g -Wall -Wextra -Werror -Wformat-security -Wfloat-equal -Wshadow -Wconversion -Wlogical-not-parentheses -Wnull-dereference -Wno-unused-variable -Wno-unused-parameter
SRC = scripts/main.c

all: clean prep compile check

clean:
  rm -rf build

prep:
  mkdir build

compile: main.bin

main.bin: $(SRC)
  $(CC) $(C_OPTS) $< -o ./build/$@

run: clean prep compile
  ./build/main.bin

check:
  clang-format --verbose --dry-run --Werror scripts/*
  clang-tidy $(SRC) -checks=-readability-uppercase-literal-suffix,-readability-magic-numbers,-clang-analyzer-deadcode.DeadStores,-clang-analyzer-security.insecureAPI.rand
  rm -rf scripts/*.dump
```

Вигляд readmi.md файлу:

```
* @mainpage
### Student : Кожевніков Ілля
### Group: КН-923в,
### Task: Лабораторна робота номер 8.2 "Вступ до документації коду"

<br />

### Загальне завдання:
Перетворити число (максимальнезначення якого - 9999) в рядок. (усі символи нижнього регістру на виході). Наприклад:
* 123 - "one hundred twenty three",
* 4311 - "four thousands three hundreds eleven

#### Розробити:
Функцію яка відображає у консолі номер у форматі слів, та в якості додаткового завдання функцію, що визначає, скільки се

#### Опис роботи основної функції:

1. Отримуємо випадкове "int" число з функції getRandom();
   Відображаємо результат printf();
2. Виконуємо функцію taskFunction() яка визначає, скільки серед заданої послідовності чисел таких пар, у котрих перше число
3. Перетворюємо номер до масиву чисел формату 1234 => []{1,2,3,4} за допомогою функції numberToIntArray;
4. Отримуємо кількість чисел у згенерованому номері за допомогою функції getNumberDigitsCount;
5. Отримуємо число з масиву за допомогою функції combineArrayToInt;
6. Генеруємо строку яка описує число за допомогою функції numberToCharArray;
7. Виводимо результат операції у консоль printf();
8. Звільняємо пам'ять масивів за допомогою функції free();

#### Перелік констант:

- 'units[]' - Масив рядків для словесного представлення чисел;
- 'teens[]' - Масив рядків для словесного представлення чисел від 11 до 19;
- 'tens[]' - Масив рядків для словесного представлення десятків;
- 'hundreds[]' - Масив рядків для словесного представлення сотень і тисяч;
- 'maxNumbersCount' - Кількість чисел у номері який можливо словесно представити.

#### Дослідження результатів роботи програми:

- 'numberAsString' - Словесне представлення числа.
- 'lab7result' - скільки серед заданої послідовності чисел таких пар, у котрих перше число менше наступного.

#### Функціонал програми:

- Генерація випадкового числа на основі часу.
- Визначення, скільки серед послідовності чисел таких пар, у котрих перше число менше наступного.
- Словесне представлення 40х значного числа.
```

Student : Кожевніков Ілля

Group: КН-923в,

Task: Лабораторна робота номер 8.2 *"Вступ до документації коду"*

Загальне завдання:

Перетворити число (максимальнезначення якого - 9999) в рядок. (усі символи нижнього регістру на виході). Наприклад:

- 123 -- "one hundred twenty three",
- 4311 -- "four thousands three hundreds eleven

Розробити:

Функцію яка відображає у консолі номер у форматі слів, та в якості **додаткового завдання** функцію, що визначає, скільки серед заданої послідовності чисел таких пар, у котрих перше число менше наступного.

Опис роботи основної функції:

1. Отримуємо випадкове int число з функції getRandom(); Відображаємо результат printf();
2. Виконуємо функцію taskFunction() яка визначає, скільки серед заданої послідовності чисел таких пар, у котрих перше число менше наступного;
3. Перетворюємо номер до масиву чисел формату 1234 => []{1,2,3,4} за допомогою функції numberToIntArray;
4. Отримуємо кількість чисел у згенерованому номері за допомогою функції getNumberDigitsCount;
5. Отримуємо число з масиву за допомогою функції combineArrayToInt;
6. Генеруємо строку яка описує число за допомогою Функції numberToCharArray;
7. Виводимо результат операції у консоль printf();
8. Звільняємо пам'ять масивів за допомогою функції free();

Перелік констант:

- units[] - Масив рядків для словесного представлення чисел;
- teens[] - Масив рядків для словесного представлення чисел від 11 до 19;
- tens[] - Масив рядків для словесного представлення десятків;
- hundreds[] - Масив рядків для словесного представлення сотень і тисяч;
- maxNumbersCount - Кількість чисел у номері який можливо словесно представити.

Дослідження результатів роботи програми:

- numberAsString - Словесне представлення числа.
- lab7result - скільки серед заданої послідовності чисел таких пар, у котрих перше число менше наступного.

Функціонал програми:

- Генерація випадкового числа на основі часу.
- Визначення, скільки серед послідовності чисел таких пар, у котрих перше число менше наступного.
- Словесне представлення 40х значного числа.

### Код основної 'main.c' операції:

```
/*
 * Функція запуску
 */
int main()
{
#define maxNumbersCount 4

    int number = getRandom();
    printf("Generated random number: %d \n", number);

    int lab7result = taskFunction(number);
    printf("\nCurrent less next iteration count: %d \n", lab7result);

    int *numberArray = numberToIntArray(number, maxNumbersCount); //ініціалізація масиву

    int numberDecimalCount = getNumberDigitsCount(number);
    int generatedPrintableNumber = combineArrayToInt(numberArray, maxNumbersCount);
    char *numberAsString = numberToCharArray(numberArray);

    printf("Randomly generated number %d. \nWrite as: %s \n", generatedPrintableNumber, numberAsString);

    free(numberArray);
    free(numberAsString);
    return 0;
}
```

### Висновки

- При виконанні лабораторної роботи були набуті практичні навички розробки програм на мові C, використання функцій, робота з масивами та циклами, а також структурування проекту.

### Автор

Кожевніков Ілля  
КН-923в

Функціонал програми:

- Генерація випадкового числа на основі часу.
- Визначення, скільки серед послідовності чисел таких пар, у котрих перше число менше наступного.
- Словесне представлення 40х значного числа.

Код основної main.c операції:

```
/**
 * Функція запуску
 */
int main()
{
#define maxNumbersCount 4

    int number = getRandom();
    printf("Generated random number: %d \n", number);

    int lab7result = taskFunction(number);
    printf("\nCurrent less next iteration count: %d \n", lab7result);

    int *numberArray = numberToIntArray(number, maxNumbersCount); //ініціалізація масиву

    int numberDecimalCount = getNumberDigitsCount(number);
    int generatedPrintableNumber = combineArrayToInt(numberArray, maxNumbersCount);
    char *numberAsString = numberToCharArray(numberArray);

    printf("Randomly generated number %d. \nWrite as: %s \n", generatedPrintableNumber, numberAsString);

    free(numberArray);
    free(numberAsString);
    return 0;
}
```

Висновки

- При виконанні лабораторної роботи були набуті практичні навички розробки програм на мові C, використання функцій, робота з масивами та циклами, а також структурування проекту.

Автор

Кожевніков Ілля  
КН-923в

## Висновки:

- При виконанні лабораторної роботи були набуті практичні навички розробки програм на мові C, використання функцій, робота з масивами та циклами, а також структурування проекту.