

**Міністерство освіти і науки України**  
**Львівський національний університет імені Івана Франка**  
**Факультет електроніки та комп'ютерних технологій**

**Звіт**

**Про виконання лабораторної роботи №9**  
**“Стрічки”**

**Виконав:**

**Студент групи ФЕП-11с**

**Качмар Денис**

**Викладач:**

**Кужій Юрій Іванович**

**Львів-2024**

**Мета:** Вивчити поняття і застосування стрічок.

### Хід роботи

- 1) Написав програму, яка реалізує введення із консолі прислів'я або афоризму із 7+ слів однією стрічкою, розбиває її на масив окремих слів із зворотнім порядком символів в словах і виводить їх в окремих рядках.

```
int main() {
    char input[MAX_LEN];
    char *words[MAX_WORDS];
    int wordCount = 0;

    printf("Введіть прислів'я або афоризм (7+ слів): ");
    fgets(input, MAX_LEN, stdin);

    char *token = strtok(input, " \n");
    while (token != NULL) {
        words[wordCount++] = token;
        token = strtok(NULL, " \n");
    }

    for (int i = 0; i < wordCount; i++) {
        reverseWord(words[i]);
        printf("%s\n", words[i]);
    }

    return 0;
}
```

```
Введіть прислів'я або афоризм (7+ слів): Lorem ipsu
m dolor sit amet, consectetur adipiscing elit
merol
muspi
rolod
tis
,tema
rutetcesnoc
gnicsipida
tile
```

- 2) Написав програму, яка реалізує пострічкове введення із консолі списку Вашої групи (улюбленої футбольної команди, будь-якого колективу з 10+ учасників) в форматі Номер-[пробіл]-Прізвище-[пробіл]-Ім'я, а потім 1) підраховує і виводить кількість відмінних (різних) імен та середню довжину прізвищ; 2) формує і виводить в алфавітному порядку список виду Номер-[пробіл]-Ім'я-[пробіл]-Прізвище.

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

#define MAX_PARTICIPANTS 100
#define MAX_LENGTH 100

typedef struct {
    int number;
    char surname[MAX_LENGTH];
    char name[MAX_LENGTH];
} Participant;

int compare(const void *a, const void *b) {
    Participant *p1 = (Participant *)a;
    Participant *p2 = (Participant *)b;
    return strcmp(p1->name, p2->name);
}

int main() {
    Participant participants[MAX_PARTICIPANTS];
    char input[MAX_LENGTH];
    int count = 0;
    int nameCount = 0;
    char names[MAX_PARTICIPANTS][MAX_LENGTH];
    int totalSurnameLength = 0;

    printf("Введіть список учасників у форматі 'Номер - Прізвище - Ім'я' (10+ учасників, введіть 'стоп' для завершення):\n");

    while (count < MAX_PARTICIPANTS) {
        fgets(input, MAX_LENGTH, stdin);
        if (strcmp(input, "стоп", 4) == 0) {
            break;
        }
        sscanf(input, "%d - %s - %s", &participants[count].number, participants[count].surname, participants[count].name);
        totalSurnameLength += strlen(participants[count].surname);

        // Додаємо ім'я до списку, якщо його ще немає
        int unique = 1;
        for (int i = 0; i < nameCount; i++) {
            if (strcmp(names[i], participants[count].name) == 0) {
                unique = 0;
                break;
            }
        }
        if (unique) {
            strcpy(names[nameCount++], participants[count].name);
        }
        count++;
    }

    double averageSurnameLength = (double)totalSurnameLength / count;

    printf("\nКількість відмінних імен: %d\n", nameCount);
    printf("Середня довжина прізвищ: %.2f\n", averageSurnameLength);

    qsort(participants, count, sizeof(Participant), compare);

    printf("\nСписок учасників в алфавітному порядку:\n");
    for (int i = 0; i < count; i++) {
        printf("%d - %s - %s\n", participants[i].number, participants[i].name, participants[i].surname);
    }

    return 0;
}
```

```
Введіть список учасників у форматі 'Номер - Прізвище - Ім'я' (10+ учасників, введіть '
стоп' для завершення):
1 - Коваленко - Олексій
2 - Петров - Андрій
3 - Іваненко - Марія
4 - Сидоренко - Дмитро
5 - Шевченко - Тарас
6 - Мельник - Анастасія
7 - Кравченко - Григорій
8 - Лисенко - Світлана
9 - Ткаченко - Олена
10 - Бондаренко - Сергій
стоп

Кількість відмінних імен: 10
Середня довжина прізвищ: 16.20

Список учасників в алфавітному порядку:
6 - Анастасія - Мельник
2 - Андрій - Петров
7 - Григорій - Кравченко
4 - Дмитро - Сидоренко
3 - Марія - Іваненко
1 - Олексій - Коваленко
9 - Олена - Ткаченко
8 - Світлана - Лисенко
10 - Сергій - Бондаренко
5 - Тарас - Шевченко
```

- 3) Написав програму, яка перевіряє на правильність введену із консолі адресу електронної пошти. Правильний формат: префікс@домен1.домен2[.домен3]. Приклади: 1) правильні – johndeer@gmail.com, ivan.franko@lnu.edu.ua; 2) неправильні – jackdoe@gmail, [Іван2020@укр.нет](mailto:Іван2020@укр.нет).

```
#include <stdio.h>
#include <string.h>
#include <stdbool.h>

bool isValidEmail(const char *email) {
    const char *at = strchr(email, '@');
    if (at == NULL || at == email || *(at + 1) == '\0') {
        return false;
    }
    const char *domain = at + 1;
    const char *dot1 = strchr(domain, '.');
    if (dot1 == NULL || dot1 == domain || *(dot1 + 1) == '\0') {
        return false;
    }
    const char *dot2 = strchr(dot1 + 1, '.');
    if (dot2 != NULL && *(dot2 + 1) == '\0') {
        return false;
    }
    if (dot2 != NULL) {
        if (dot1 + 1 == dot2 || *(dot2 + 1) == '\0') {
            return false;
        }
    }
    for (const char *ptr = email; ptr < at; ptr++) {
        if (*ptr == ' ' || *ptr == '@' || *ptr == '.') {
            return false;
        }
    }
    for (const char *ptr = domain; *ptr != '\0'; ptr++) {
        if (*ptr == ' ' || *ptr == '@') {
            return false;
        }
    }
    return true;
}

int main() {
    char email[100];
    printf("Введіть адресу електронної пошти: ");
    scanf("%s", email);
    if (isValidEmail(email)) {
        printf("Адреса електронної пошти правильна.\n");
    } else {
        printf("Адреса електронної пошти неправильна.\n");
    }
    return 0;
}
```

```
Введіть адресу електронної пошти: Іван2020@укр.нет
Адреса електронної пошти правильна.
```

**Висновок:** Протягом виконання завдань я створив три програми на мові С, які допомагають у вирішенні різних задач. Перша програма обробляє прислів'я або афоризми, розбиваючи їх на слова та реверсуючи символи в кожному слові. Друга програма реалізує введення списку учасників групи, підраховує унікальні імена та середню довжину прізвищ, а також формує алфавітний список. Третя програма перевіряє правильність введених адрес електронної пошти відповідно до заданого формату. Ці завдання дозволили мені закріпити навички програмування, працювати з масивами та стрічками, а також покращити логіку вирішення задач.