

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-11
Захарчук Адріан

Перевірив:
ас. Кужій Ю.І

Мета роботи:

Вивчити поняття і застосування стрічок

Обладнання та програмне забезпечення:

- IBM сумісна персональна обчислювальна машина
- Онлайн компілятор мови програмування Cі, доступний за посиланням <https://repl.it/languages/c>

Завдання до роботи:

Написати програму на мові програмування Cі, котра використовує наступні дії над стрічками:

- пошук заданої послідовності символів;
- заміну і видалення символів і груп символів;
- порівняння і сортування стрічок;
- знаходження кількості символів у стрічках.

ПОРЯДОК ВИКОНАННЯ РОБОТИ:

1. Опрацювати і засвоїти матеріал наведений в теоретичних відомостях.
2. Написати 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.
3. Написати 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.
4. Написати 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.

5. В звіті навести копії екранів та написаний код.
6. Зробити висновки.

2. Написати 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.

```
1 #include <stdio.h>
2 #include <string.h>
3
4 int countWords(const char *string) {
5     int count = 0;
6     const char *ptr = string;
7     while (*ptr) {
8         if (*ptr == ' ' || *ptr == '\n')
9             count++;
10        ptr++;
11    }
12    return count;
13 }
14
15 void reverseAndPrintWord(const char *word) {
16     for (int i = strlen(word) - 1; i >= 0; --i) {
17         printf("%c", word[i]);
18     }
19     printf("\n");
20 }
21
22 int main(void) {
23     char string[100];
24     printf("Enter a proverb or aphorism with at least 7 words in English: ");
25     fgets(string, sizeof(string), stdin);
26
27     if (countWords(string) < 7) {
28         printf("The sentence must contain at least 7 words.\n");
29         return 0;
30     }
31
32     printf("Reversed words:\n");
33     char *word = strtok(string, " \n");
34     while (word != NULL) {
35         reverseAndPrintWord(word);
36         word = strtok(NULL, " \n");
37     }
38
39     return 0;
40 }
```

Run

Enter a proverb or aphorism with at least 7 words in English: lets wander where the wi-fi is weak
Reversed words:
stel
rednaw
erehw
eht
if-iw
si
kaew

3. Написати 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.

```
task2.c > () anon struct Participant > ...
1 #include <stdio.h>
2 #include <string.h>
3 #include <stdlib.h>
4
5 #define MAX_NAME_LENGTH 50
6
7 typedef struct {
8     int number;
9     char surname[MAX_NAME_LENGTH];
10    char name[MAX_NAME_LENGTH];
11 } Participant;
12
13 void sortParticipants(Participant participants[], int count) {
14     for (int i = 0; i < count - 1; i++) {
15         for (int j = i + 1; j < count; j++) {
16             if (strcmp(participants[i].name, participants[j].name) > 0) {
17                 Participant temp = participants[i];
18                 participants[i] = participants[j];
19                 participants[j] = temp;
20             }
21         }
22     }
23 }
24
25 int countUniqueNames(Participant participants[], int count) {
26     char uniqueNames[count][MAX_NAME_LENGTH];
27     int uniqueCount = 0;
28
29     for (int i = 0; i < count; i++) {
30         int found = 0;
31         for (int j = 0; j < uniqueCount; j++) {
32             if (strcmp(participants[i].name, uniqueNames[j]) == 0) {
33                 found = 1;
34                 break;
35             }
36         }
37         if (!found) {
38             strcpy(uniqueNames[uniqueCount], participants[i].name);
39             uniqueCount++;
40         }
41     }
42     return uniqueCount;
43 }
44
45 double calculateAverageSurnameLength(Participant participants[], int count) {
46     int totalLength = 0;
47     for (int i = 0; i < count; i++) {
48         totalLength += strlen(participants[i].surname);
49     }
50     return (double)totalLength / count;
51 }
52
53 int main() {
54     int numParticipants;
55
56     printf("Enter the number of participants (at least 10): ");
57     scanf("%d", &numParticipants);
58
59     if (numParticipants < 10) {
60         printf("The number of participants must be at least 10.\n");
61         return 1;
62     }
63 }
```

Run

Enter the number of participants (at least 10): 10
Enter the list of participants in the format (Number Surname Name):
Participant 1: 1 adrian zakharchuk
Participant 2: 2 taras manachynskiy
Participant 3: 3 ostap zinchuk
Participant 4: 4 nazar levandivskiy
Participant 5: 5 daniilo garbich
Participant 6: 6 ivan germanovich
Participant 7: 7 nazar potseluiko
Participant 8: 8 denys kachmar
Participant 9: 9 misha dorazh
Participant 10: 10 antoshkova alina

Number of unique names: 10
Average surname length: 5.00

Sorted participants (Number-Name-Surname):
10 alina antoshkova
9 dorazh misha
3 garbich daniilo
6 germanovich ivan
8 kachmar denys
4 levandivskiy nazar
2 manachynskiy taras
7 potseluiko nazar
1 zakharchuk adrian
3 zinchuk ostap

Ln 7, Col 17 • Spaces: 4 History

```

02     }
03
04     Participant participants[numParticipants];
05
06     printf("Enter the list of participants in the format (Number Surname Name)
07     \n");
08     for (int i = 0; i < numParticipants; i++) {
09         printf("Participant %d: ", i + 1);
10         scanf("%d %s %s", &participants[i].number, participants[i].surname,
11             participants[i].name);
12     }
13
14     // 1) Calculate unique names and average surname length
15     int uniqueNames = countUniqueNames(participants, numParticipants);
16     double avgSurnameLength = calculateAverageSurnameLength(participants,
17         numParticipants);
18
19     printf("\nNumber of unique names: %d\n", uniqueNames);
20     printf("Average surname length: %.2f\n", avgSurnameLength);
21
22     // 2) Sort and display participants
23     sortParticipants(participants, numParticipants);
24
25     printf("\nSorted participants (Number-Name-Surname):\n");
26     for (int i = 0; i < numParticipants; i++) {
27         printf("%d %s %s\n", participants[i].number, participants[i].name,
28             participants[i].surname);
29     }
30
31     return 0;
32 }
33
34

```

4. Написати 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.

```

1  #include <regex.h>
2  #include <stdio.h>
3
4  int validateEmail(const char *email) {
5      const char *pattern =
6          "[a-zA-Z0-9-._%+~]+@[a-zA-Z0-9-]+\.[a-zA-Z]{2,}(\.[a-zA-Z]{2,})?";
7      regex_t regex;
8
9      if (regcomp(&regex, pattern, REG_EXTENDED) != 0) {
10         printf("Could not compile regex.\n");
11         return 0;
12     }
13
14     int status = regexec(&regex, email, 0, NULL, 0);
15     regfree(&regex);
16
17     return status == 0;
18 }
19
20 int main() {
21     char email[256];
22
23     printf("Enter an email address: ");
24     scanf("%s", email);
25
26     if (validateEmail(email)) {
27         printf("Valid email address.\n");
28     } else {
29         printf("Invalid email address.\n");
30     }
31
32     return 0;
33 }
34

```

```

Enter an email address: adrian@gmail
Invalid email address.

```

Висновок: під час виконання цієї лабораторної роботи я освоїв роботу зі строками. Зокрема, я розробив кілька програм: для введення прислів'я та поділу його на масив окремих слів, для введення списку учасників групи з подальшим сортуванням, а також для перевірки правильності введеної електронної адреси.