

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ  
УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**

**Лабораторна робота з дисципліни  
«Алгоритмізація та програмування»**

**Виконав:**  
**студент групи КН-109**  
**Регусевич Теодор**  
**Викладач:**  
**Варецький Я.**

**Львів – 2018 р.**

## Лабораторна робота № 9.

**Тема:** "Ввід-вивід рядків"

**Мета:** Робота з текстовими файлами, ввід-вивід текстової інформації і її зберігання на зовнішніх носіях.

### 2. Постановка завдання

Створити текстовий файл F1 не менше, ніж з 10 рядків і записати в нього інформацію

Виконати завдання.

### Варіант 24

- 1) Скопіювати з файлу F1 у файл F2 всі рядки, у яких міститься два однакових слова.
- 2) Визначити номер слова, у якому найбільше букв «А».

### Код програми:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
void formfile (FILE *f) {
    char buffer[255];
    printf("Enter 5 strings to fill f1.bin: \n");
    for (int i=0; i<5; i++) {
        fgets(buffer,255,stdin);
        fputs(buffer, f);
        if (ferror(f)) {
            printf("Error while writing!\n");
            exit(1);
        }
    }
    rewind(f);
};
void showfile (FILE *f) {
    char buffer[255];
    char buffer2[255];
    int i=0;
    if (f) {
        while (fscanf(f, "%s", buffer) != EOF)
            printf("%s\n", buffer);
    }
    rewind(f);
};
void copy(FILE *f, FILE *f2) {
    char *token;
    char *word;
    char *buffer[255];
    char *buffer3[255];
    char buffer2[255];
    int i = 0, counter = 0, n = 0, m = 0, word_numbeer = 0, checker=0;
```

```

char *str[100];
if (f) {
    while (fgets(buffer2,255,f)) {
        buffer[n] = strdup(buffer2);
        buffer3[n] = strdup(buffer2);
        token = strtok(buffer2, " ");
        while (token != NULL) {
            str[i] = strdup(token);
            token = strtok(NULL, " ");
            i++;
            counter++;
        }
        n++;
    }
    i=0;
    printf("\n\n\n");
}
int number[n];
for (int j = 0; j < counter; ++j) {
    for (int k = 0; k < n; ++k) {
        for (int l = 0; l < n; ++l) {
            buffer[l] = strdup(buffer3[l]);
        }
        for (int l = 0; l < n; ++l) {
            if(number[l]==1)
            {
                k++;
            }
        }
        token=strtok(buffer[k], " ");
        while (token!=NULL)
        {
            if(strcmp(token,str[j])==0)
            {
                m++;
            }
            if(m>=2)
            {
                number[k]=1;
                word = str[j];
                printf("%s",buffer3[k]);
                fputs(buffer3[k],f2);
                checker=1;
                break;
            }
            token=strtok(NULL, " ");
        }
        m=0;
        if (checker==1)
        {
            checker=0;
            break;
        }
    }
}
}

void find_A(FILE *f) {
    char *token;
    char buffer[255];
    int i=0,counter=0,n=0,max=0,word_numbeer=0;
    char *str[100];
    if (f) {
        while (fscanf(f, "%s", buffer) != EOF) {
            token = strtok(buffer, " ");

```

```

        while (token != NULL) {
            str[i] = strdup(token);
            counter++;
            for (int j = 0; j < strlen(token); ++j) {
                if(token[j]=='a' || token[j]=='A')
                {
                    n++;
                    if(max<n)
                    {
                        max=n;
                        word_numbeer=counter;
                    }
                }
            }
            n=0;
            i++;
            token = strtok(NULL, " ");
            printf("%s\n", str[i]);*/
            i++;
        }
        printf("%d - %d",max,word_numbeer);
    }
}

int main() {
    FILE *file1;
    FILE *file2;
    file1 = fopen("gg.txt","w");
    formfile(file1);
    fclose(file1);
    file1 = fopen("gg.txt","r");
    file2 = fopen("ggg.txt","w");
    showfile(file1);
    copy(file1,file2);
    fclose(file2);
    fopen("ggg.txt","r");
    showfile(file2);
    find_A(file2);
    fclose(file1);
    fclose(file2);
    return 0;
}

```

## Результат:

```
Enter 5 strings to fill f1.bin:
aa aa dsa sa
dssdssdsds
dasdsadasasd
ss sss ss aaaaaaaaaa
dsadsadsasdads

aa
aa
dsa
sa
dssdssdsds
dasdsadasasd
ss
sss
ss
aaaaaaaaa
dsadsadsasdads


aa aa dsa sa
ss sss ss aaaaaaaaaa
aa
aa
dsa
sa
ss
sss
ss
aaaaaaaaa
9 - 8
Process finished with exit code 0
```