

Харківський національний університет імені В. Н. Каразіна Факультет
комп'ютерних наук
Кафедра штучного інтелекту та програмного забезпечення

ЗВІТ
Практична робота №1
дисципліна: «Бази даних»

Виконав: студент групи КС-22
Ковальов Андрій
Перевірив: викладач
Бережний Артем

Харків

Мета роботи: Створити систему, яка дозволяє користувачу, виконати зчитування з файлу, виконувати пошук за відповідним критерієм або відображати зальну інформацію яку було зчитано.

Завдання

Вступ

В наші дні все стає оцифрованим, від рутинних завдань до важливих, і ми побачили, що бібліотека в UI повинна бути онлайн і мати більше функцій, тому, коли нас попросили вибрати ідею і написати про неї виклад проблеми, ми подивилися навколо - у університеті - щоб побачити, чи зможемо ми вловити якесь натхнення з цього, після того як знайшли ми почали розуміти, що для того, щоб взяти книгу або купити книгу в бібліотеці, ви повинні бути присутніми в бібліотеці, ви не можете цього зробити онлайн, тому ми подумали, що якщо немає цифрової системи, яка дозволяла б студенту купувати, позичати або показувати статус книги, якщо вона доступна чи ні, без втрати часу через онлайн-систему.

Підготовка до завдання

Це лабораторія **парного програмування**. Ви несете відповідальність за пошук своїх товаришів по команді для цього завдання, і вам потрібно робити це швидко. Я призначу партнерів для всіх, хто не зміг їх знайти. Це лабораторне завдання повинно зайняти близько 2 годин вашого лабораторного часу. З цієї причини крайній термін для цієї частини – **2 тижня** (з моменту видачі).

Пропонована система

Пропонована система, яку необхідно розробити – це система, яка дозволяє користувачу, виконати зчитування з файлу, виконувати пошук за відповідним критерієм або відображати зальну інформацію яку було зчитано. Роботу необхідно зробити на об'єктно-орієнтованій мові програмування, яка найбільш зручна для Вас. **Використання сторонніх бібліотек не рекомендовано.** У даній роботі не треба використовувати БД, та підходи які автоматично генерують структуру додатку.

Вимоги до розробки

Hardware	Software
-Processor: i3 -Speed: 2.5 GHz -Hard Disk: 500MB -Ram: 2GB	-Operating system: Windows OS -Language: Java, Python, C#, XMAL, JSON, XML -IDE: Visual Studio або відповідна

The Task

Завдання складається з двох частин. Перша частина завдання дається вам зараз, друга частина буде надана команді, як тільки команда повідомить про виконання першого завдання. Вам надається список учнів місцевої початкової школи разом із їх класом. Список зберігається у файлі **students.txt** (його можна скачати з **stm.khai.edu**). У кожному рядку файлу зберігається інформація про студента.

Формат рядка:

StLastName, StFirstName, Grade, Classroom, Bus, TLastName, TFirstName

Тут, StLastName і StFirstName ідентифікують студента; TFirstName ідентифікує вчителя учня. Bus - це маршрут шкільного автобуса, яким учень їде, щоб добратися до школи. Bus, Grade і Classroom - цілі числа (якщо значення цього поля 0, то це означає -**дитячий садок**, учень, який самостійно прибуває до школи – значення відмінне від 0), а всі інші поля - це рядки.

Ось зразок рядка з файлу:

DROP, SHERMAN, 0, 104, 51, NIBLER, JERLENE

Не дивно, що рядок слід читати «Sherman Drop, який їде автобусом 51 маршрутом, є вихованцем дитячого садка, Mrs. Jerlene Nibler у класі 104».

Ваша мета - написати програму, яка здійснює пошук студентів у текстовому файлі та видає результати пошуку.

Потрібно здійснити такі пошукові запити:

- Враховуючи прізвище студента, знайдіть клас студента, та клас викладача (якщо є більше одного студента з однаковим прізвищем, знайдіть цю інформацію для всіх студентів);
- Враховуючи прізвище студента, знайдіть автобусний маршрут, яким їде студент (якщо є більше одного студента з однаковим прізвищем, знайдіть цю інформацію для всіх студентів);
- Дан викладач, знайдіть список його учнів;
- Враховуючи автобусний маршрут, знайдіть усіх учнів, які ним користуються;
- Знайдіть усіх учнів на певному рівні класу;

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

#define MAX_LINE_LEN 100
#define MAX_STUDENTS 10000000

typedef struct {
    char lastName[30];
    char firstName[30];
    int grade;
    int classroom;
    int bus;
    char teacherLastName[30];
    char teacherFirstName[30];
} Student;

Student students[MAX_STUDENTS];
int studentCount = 0;

void loadStudents(const char *filename) {
```

```

FILE *file = fopen(filename, "r");
if (file == NULL) {
    printf("Не вдалося відкрити файл: %s\n", filename);
    exit(1);
}

while (studentCount < MAX_STUDENTS && fscanf(file, "%[^,], %[^,], %d, %d, %d, %[^,], %[^\\n]\\n",
        students[studentCount].lastName,
        students[studentCount].firstName,
        &students[studentCount].grade,
        &students[studentCount].classroom,
        &students[studentCount].bus,
        students[studentCount].teacherLastName,
        students[studentCount].teacherFirstName) == 7) {
    studentCount++;
}

fclose(file);
}

void searchStudentByLastName(char *lastName) {
    int found = 0;
    for (int i = 0; i < studentCount; i++) {
        if (strcmp(students[i].lastName, lastName) == 0) {
            printf("Студент: %s %s, Клас: %d, Клас викладача: %d, Викладач: %s %s\n",
                students[i].lastName, students[i].firstName,
                students[i].classroom, students[i].classroom,
                students[i].teacherLastName, students[i].teacherFirstName);
            found = 1;
        }
    }
    if (!found) {
        printf("Студент з прізвищем %s не знайдений.\n", lastName);
    }
}

void searchStudentBus(char *lastName) {
    int found = 0;
    for (int i = 0; i < studentCount; i++) {
        if (strcmp(students[i].lastName, lastName) == 0) {
            printf("Студент: %s %s, Маршрут автобуса: %d\n",
                students[i].lastName, students[i].firstName, students[i].bus);
            found = 1;
        }
    }
    if (!found) {
        printf("Студент з прізвищем %s не знайдений.\n", lastName);
    }
}

```

```

    }
}

void searchTeacherStudents(char *lastName) {
    int found = 0;
    for (int i = 0; i < studentCount; i++) {
        if (strcmp(students[i].teacherLastName, lastName) == 0) {
            printf("Студент: %s %s\n", students[i].lastName, students[i].firstName);
            found = 1;
        }
    }
    if (!found) {
        printf("Учнів у викладача з прізвищем %s не знайдено.\n", lastName);
    }
}

void searchByClassroom(int classroom) {
    int found = 0;
    for (int i = 0; i < studentCount; i++) {
        if (students[i].classroom == classroom) {
            printf("Студент: %s %s\n", students[i].lastName, students[i].firstName);
            found = 1;
        }
    }
    if (!found) {
        printf("Студенти у класі номер %d не знайдені.\n", classroom);
    }
}

void searchByBusRoute(int bus) {
    int found = 0;
    for (int i = 0; i < studentCount; i++) {
        if (students[i].bus == bus) {
            printf("Студент: %s %s, Клас: %d, Клас викладача: %d\n",
                students[i].lastName, students[i].firstName, students[i].classroom, students[i].classroom);
            found = 1;
        }
    }
    if (!found) {
        printf("Студенти на маршруті автобуса номер %d не знайдені.\n", bus);
    }
}

int main() {
    loadStudents("students.txt");

    char command[20];
    clock_t start, end;

```

```
double cpu_time_used;

while (1) {
    printf("\nКоманди: S[tudent], SB[us], T[eacher], C[lassroom], B[us], Q[uit]. ");
    printf("\nВведіть команду: ");
    scanf("%s", command);

    if (strcmp(command, "Q") == 0) {
        break;
    } else if (strncmp(command, "S", 7) == 0) {
        char lastName[30];
        scanf("%s", lastName);

        start = clock();

        searchStudentByLastName(lastName);
        end = clock();
    } else if (strncmp(command, "SB", 8) == 0) {
        char lastName[30];
        scanf("%s", lastName);

        start = clock();
        searchStudentBus(lastName);
        end = clock();
    }
    else if (strncmp(command, "T", 7) == 0) {
        char lastName[30];
        scanf("%s", lastName);

        start = clock();
        searchTeacherStudents(lastName);
        end = clock();
    } else if (strncmp(command, "C", 9) == 0) {
        int classroom;
        scanf("%d", &classroom);

        start = clock();
        searchByClassroom(classroom);
        end = clock();
    } else if (strncmp(command, "B", 3) == 0) {
        int bus;
        scanf("%d", &bus);

        start = clock();
        searchByBusRoute(bus);
        end = clock();
    } else {
        printf("Невідома команда.\n");
        continue;
    }
}
```

```

    }

    // Підрахунок часу пошуку
    cpu_time_used = ((double) (end - start)) / CLOCKS_PER_SEC;
    printf("Час пошуку: %.6f секунд\n", cpu_time_used);
}

return 0;
}

```

Лістинг 1 – код програми

```

Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].
Введіть команду: S COOKUS

```

```

Студент: COOKUS XUAN, Клас: 105, Клас викладача: 105, Викладач: BODZIONY JED
Студент: COOKUS INGER, Клас: 103, Клас викладача: 103, Викладач: HANTZ BENITO
Студент: COOKUS DICK, Клас: 104, Клас викладача: 104, Викладач: HANTZ BENITO
Студент: COOKUS MANIE, Клас: 102, Клас викладача: 102, Викладач: GAMBREL NANCY
Студент: COOKUS TAMESHA, Клас: 104, Клас викладача: 104, Викладач: HANTZ JAE
Студент: COOKUS BILLY, Клас: 111, Клас викладача: 111, Викладач: HAMER JED
Студент: COOKUS BILLY, Клас: 111, Клас викладача: 111, Викладач: STEIB NANCY
Студент: COOKUS BILLY, Клас: 109, Клас викладача: 109, Викладач: STEIB JAE
Студент: COOKUS XUAN, Клас: 107, Клас викладача: 107, Викладач: FAFARD ROCIO
Студент: COOKUS XUAN, Клас: 104, Клас викладача: 104, Викладач: KERBS JONATHAN
Студент: COOKUS MANIE, Клас: 112, Клас викладача: 112, Викладач: COOL ROCIO
Студент: COOKUS WAN, Клас: 109, Клас викладача: 109, Викладач: FAFARD GAVIN
Студент: COOKUS RANDOLPH, Клас: 111, Клас викладача: 111, Викладач: HANTZ PERLA
Студент: COOKUS JANNETTE, Клас: 105, Клас викладача: 105, Викладач: NISTENDIRK ADOLPH
Студент: COOKUS INGER, Клас: 110, Клас викладача: 110, Викладач: GAMBREL REUBEN
Студент: COOKUS TOMAS, Клас: 109, Клас викладача: 109, Викладач: CHIONCHIO BENITO
Студент: COOKUS ZANDRA, Клас: 106, Клас викладача: 106, Викладач: FAFARD REUBEN
Студент: COOKUS HYE, Клас: 107, Клас викладача: 107, Викладач: CHIONCHIO JONATHAN
Студент: COOKUS XUAN, Клас: 105, Клас викладача: 105, Викладач: BODZIONY JED
Студент: COOKUS INGER, Клас: 103, Клас викладача: 103, Викладач: HANTZ BENITO
Студент: COOKUS DICK, Клас: 104, Клас викладача: 104, Викладач: HANTZ BENITO
Студент: COOKUS MANIE, Клас: 102, Клас викладача: 102, Викладач: GAMBREL NANCY
Студент: COOKUS TAMESHA, Клас: 104, Клас викладача: 104, Викладач: HANTZ JAE
Студент: COOKUS BILLY, Клас: 111, Клас викладача: 111, Викладач: HAMER JED
Студент: COOKUS BILLY, Клас: 111, Клас викладача: 111, Викладач: STEIB NANCY
Студент: COOKUS BILLY, Клас: 109, Клас викладача: 109, Викладач: STEIB JAE
Час пошуку: 2.918000 секунд

Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].
Введіть команду: 

```

Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].
Введіть команду: SB COOKUS

Студент: COOKUS XUAN, Маршрут автобуса: 52
Студент: COOKUS INGER, Маршрут автобуса: 51
Студент: COOKUS DICK, Маршрут автобуса: 54
Студент: COOKUS MANIE, Маршрут автобуса: 53
Студент: COOKUS TAMESHA, Маршрут автобуса: 53
Студент: COOKUS BILLY, Маршрут автобуса: 51
Студент: COOKUS BILLY, Маршрут автобуса: 55
Студент: COOKUS BILLY, Маршрут автобуса: 51
Студент: COOKUS XUAN, Маршрут автобуса: 52
Студент: COOKUS XUAN, Маршрут автобуса: 55
Студент: COOKUS MANIE, Маршрут автобуса: 53
Студент: COOKUS WAN, Маршрут автобуса: 56
Студент: COOKUS RANDOLPH, Маршрут автобуса: 52
Студент: COOKUS JANNETTE, Маршрут автобуса: 51
Студент: COOKUS INGER, Маршрут автобуса: 54
Студент: COOKUS TOMAS, Маршрут автобуса: 54
Студент: COOKUS ZANDRA, Маршрут автобуса: 51
Студент: COOKUS HYE, Маршрут автобуса: 56
Студент: COOKUS XUAN, Маршрут автобуса: 52
Студент: COOKUS INGER, Маршрут автобуса: 51
Студент: COOKUS DICK, Маршрут автобуса: 54
Студент: COOKUS MANIE, Маршрут автобуса: 53
Студент: COOKUS TAMESHA, Маршрут автобуса: 53
Студент: COOKUS BILLY, Маршрут автобуса: 51
Студент: COOKUS BILLY, Маршрут автобуса: 55
Студент: COOKUS BILLY, Маршрут автобуса: 51
Час пошуку: 2.771000 секунд

Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].
Введіть команду: █

Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].
Введіть команду: T STEIB JAE

Студент: CREMEANS TAMESHA
Студент: BERBES TOMAS
Студент: CORONADO CARTER
Студент: BEX HYE
Студент: CORONADO DICK
Студент: CORKER TAMESHA
Студент: COVINGTON TAMESHA
Студент: CREMEANS CARTER
Студент: BUSSMANN RANDOLPH
Студент: CLECKLER INGER
Студент: BRODERSEN MANIE
Студент: BUSSMANN TAMESHA
Студент: BREVO TOMAS
Студент: BEX DIMPLE
Студент: COVINGTON FLOY
Студент: COOKUS BILLY
Студент: BEX RANDOLPH
Студент: COOKUS BILLY
Студент: CORONADO DIMPLE
Студент: CORONADO JANNETTE
Студент: CIGANEK JANNETTE
Студент: BOYTER JANNETTE
Студент: BUSSMANN INGER
Час пошуку: 3.025000 секунд

Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].

Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].
Введіть команду: C 111

Студент: BUSSMANN TAMESHA
Студент: BUSSMANN MANIE
Студент: BEAN BILLY
Студент: COOKUS RANDOLPH
Студент: BERBES WAN
Студент: CORKER JANNETTE
Студент: CLECKLER FLOY
Студент: BEAN JANNETTE
Студент: BERBES DICK
Студент: COVINGTON TAMESHA
Студент: CREMEANS CARTER
Студент: CORKER ZANDRA
Студент: BERBES XUAN
Студент: CREMEANS CARTER
Студент: CLECKLER INGER
Студент: BERBES INGER
Студент: CORONADO HYE
Студент: COOKUS BILLY
Студент: BRODERSEN BILLY
Студент: COOKUS BILLY
Студент: CORKER HYE
Студент: CORONADO TAMESHA
Студент: BRODERSEN DICK
Студент: COVINGTON RANDOLPH
Студент: CREMEANS DICK
Студент: BOYTER WAN
Час пошуку: 3.084000 секунд

Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].
Введіть команду:

```
Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].  
Введіть команду: B 51
```

```
Студент: BUSSMANN ZANDRA, Клас: 101, Клас викладача: 101  
Студент: CORKER MANIE, Клас: 101, Клас викладача: 101  
Студент: CORKER ZANDRA, Клас: 107, Клас викладача: 107  
Студент: BOYTER INGER, Клас: 101, Клас викладача: 101  
Студент: BEX JANNETTE, Клас: 105, Клас викладача: 105  
Студент: BERBES XUAN, Клас: 112, Клас викладача: 112  
Студент: CREMEANS CARTER, Клас: 111, Клас викладача: 111  
Студент: COVINGTON ZANDRA, Клас: 106, Клас викладача: 106  
Студент: BUSSMANN TAMESHA, Клас: 101, Клас викладача: 101  
Студент: CLECKLER MANIE, Клас: 109, Клас викладача: 109  
Студент: CORONADO DIMPLE, Клас: 102, Клас викладача: 102  
Студент: BERBES WAN, Клас: 112, Клас викладача: 112  
Студент: COOKUS BILLY, Клас: 111, Клас викладача: 111  
Студент: BRODERSEN BILLY, Клас: 111, Клас викладача: 111  
Студент: COVINGTON FLOY, Клас: 109, Клас викладача: 109  
Студент: COMO RANDOLPH, Клас: 112, Клас викладача: 112  
Студент: COVINGTON JANNETTE, Клас: 102, Клас викладача: 102  
Студент: BREVO JANNETTE, Клас: 106, Клас викладача: 106  
Студент: COMO TAMESHA, Клас: 112, Клас викладача: 112  
Студент: COMO BILLY, Клас: 108, Клас викладача: 108  
Студент: CREMEANS RANDOLPH, Клас: 106, Клас викладача: 106  
Студент: COOKUS BILLY, Клас: 109, Клас викладача: 109  
Студент: CORONADO DIMPLE, Клас: 106, Клас викладача: 106  
Студент: BRODERSEN DICK, Клас: 111, Клас викладача: 111  
Студент: BEX BILLY, Клас: 103, Клас викладача: 103  
Студент: CIGANEK WAN, Клас: 104, Клас викладача: 104  
Час пошуку: 4.039000 секунд
```

```
Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].  
Введіть команду:
```

```
Команди: S[tudent], SB[us], T[eacher], C[classroom], B[us], Q[uit].  
Введіть команду: Q
```

```
-----  
Process exited after 2.495 seconds with return value 0  
Press any key to continue . . .
```

Скріншот 1-9 – виконання програми

Пояснення:

Програма здійснює пошук інформації про студентів, вчителів, класи та автобуси, що зберігається у файлі `students.txt`. Користувач може вводити команди в консоль для отримання відповідних даних.

Команди

Програма підтримує кілька типів команд, кожна з яких виконує різний пошук:

- **S <lastname>** – Пошук студента за прізвищем.
- **SB <lastname>** – Пошук студента за прізвищем та виводить маршрут шкільного автобуса.
- **T <lastname>** – Пошук студентів за викладачем.
- **C <number>** – Пошук студентів за номером класу.
- **B <number>** – Пошук студентів за автобусним маршрутом.
- **Q** – Вихід з програми.

Підсумок:

- Програма дає можливість користувачу зручно вводити та шукати студентів, вчителів та іншої інформації, обробляючи задані команди.
- Програма працює доки користувач не вийде з програми.

Висновок:

В цій практичній роботі я отримав навички роботи з файлами, обробки текстових даних та використання структур для зберігання інформації про студентів. Я також використав базові алгоритми пошуку в масивах даних. За допомогою цього досвіду я зрозумів що являє собою даний вид роботи

Посилання на GitHub: <https://github.com/yunkaa-k/bd1>