

Лабораторная 2.

Задание.

1. Написать программу для манипуляциями данных на основе рекурсивных структур, реализовать функции вставки, удаления и навигации для списка, реализовать сериализацию списка (например электронный журнал успеваемости студентов)
2. Провести трассировку программы с помощью gdb

Инструментарий: редактор vim, компилятор gcc, отладчик gdb. Язык Си

Описание работы программы

Программа реализует работу со списком студентов, используя односвязный список. Она позволяет добавлять студентов, выводить их на экран, сериализовать и десериализовать данные, а также выполнять поиск и удаление студентов по фамилии и сортировку по оценкам.

Функции программы

addStudent(const char surname[], double mark) - Добавляет нового студента в конец списка.

printList() - Выводит список студентов на экран. Если список пуст, сообщает об этом.

serialize() - Сохраняет список студентов в бинарный файл students.dat.

deserialize() - Загружает список студентов из файла students.dat.

searchInList(const char surname[]) - Ищет студента по фамилии и выводит информацию о нем, если найден.

deleteFromList(const char surname[]) - Удаляет студента из списка по фамилии. Если студент не найден, выводит соответствующее сообщение.

sortByMark() - Сортирует список студентов по оценкам с использованием алгоритма пузырьковой сортировки.

```

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

#define MAX_NAME_LENGTH 30

struct Student {
    struct Student *next;
    char surname[MAX_NAME_LENGTH];
    double mark;
};

struct Student *head = NULL;
struct Student *tail = NULL;

// добавление студента в список
void addStudent(const char surname[], double mark) {
    struct Student *newNode = (struct Student *)malloc(sizeof(struct Student));
    if (newNode == NULL) {
        fprintf(stderr, "Memory allocation failed\n");
        return;
    }
    strncpy(newNode->surname, surname, MAX_NAME_LENGTH);
    newNode->mark = mark;
    newNode->next = NULL;

    if (head == NULL) {
        head = newNode;
        tail = newNode;
    } else {
        tail->next = newNode;
        tail = newNode;
    }
}

// вывод списка студентов
void printList() {
    struct Student *current = head;

    if (current == NULL) {
        printf("The list is empty.\n");
        return;
    }

    while (current != NULL) {
        printf("Surname: %s, Mark: %.2lf\n", current->surname, current->mark);
        current = current->next;
    }
}

```

код программы открытый в редакторе vim

```
tania@TaniaLaptop:~/5sem/os/os_labs/lab2$ gcc lab2.c -o lab2
tania@TaniaLaptop:~/5sem/os/os_labs/lab2$ ./lab2
Before sorting:
Surname: Ivanov, Mark: 4.50
Surname: Petrov, Mark: 3.00
Surname: Sidorov, Mark: 5.00

After sorting:
Surname: Petrov, Mark: 3.00
Surname: Ivanov, Mark: 4.50
Surname: Sidorov, Mark: 5.00

Enter the surname to search:
Petrov
Found: Surname: Petrov, Mark: 3.00

Enter the surname to delete: Sidorov
Deleted: Sidorov

After deletion:
Surname: Petrov, Mark: 3.00
Surname: Ivanov, Mark: 4.50
tania@TaniaLaptop:~/5sem/os/os_labs/lab2$ |
```

результат работы программы

```

tania@TaniaLaptop:~/5sem/os/os_labs/lab2$ gcc -g lab2.c -o lab2
tania@TaniaLaptop:~/5sem/os/os_labs/lab2$ gdb ./lab2
GNU gdb (Ubuntu 12.1-0ubuntu1~22.04.2) 12.1
Copyright (C) 2022 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./lab2...
(gdb) break lab2.c:10
Breakpoint 1 at 0x12de: file lab2.c, line 18.
(gdb) break ma
main      malloc      malloc@plt
(gdb) break main
Breakpoint 2 at 0x1852: file lab2.c, line 159.
(gdb) run
Starting program: /home/tania/5sem/os/os_labs/lab2/lab2
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".

Breakpoint 2, main () at lab2.c:159
159     int main() {
(gdb)
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) n
Program not restarted.
(gdb)
(gdb) print
The history is empty.
(gdb) next
160         addStudent("Ivanov", 4.5);
(gdb) next

Breakpoint 1, addStudent (surname=0x5555555555612c "Ivanov", mark=4.5) at lab2.c:18
18         struct Student *newNode = (struct Student *)malloc(sizeof(struct Student));
(gdb) step
__GI___libc_malloc (bytes=48) at ./malloc/malloc.c:3287
3287     ./malloc/malloc.c: No such file or directory.
(gdb) continue

```

трассировка программы с помощью gdb