

Лабораторна робота №19

Тема: Динамічні списки.

Мета: Навчитися працювати динамічними списками.

Індивідуальне завдання

Реалізувати наступні функції роботи зі списком:

1. читання даних з файлу;
2. запис даних у файл;
3. вивід вмісту списку на екран;
4. додавання об'єкта у кінець списку;
5. видалення об'єкта зі списку.

Хід роботи

1. Створив динамічний список прикладної області. Код показано на рисунку.

```
#define FILENAME "file.dat"
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <iomanip>
#include <conio.h>
#include <Windows.h>
using namespace std;

struct Zoo
{
    char name[50];
    int number;
    char colour[50];
    char continent[50];
    char kind[50];
    int age;
};

void console_clear();
void form_file();
void read_file();
void work_file();
void delete_from_file(int startAge, int endAge, char* name);
int clear_file(const char* filename);
Zoo create();
```

Рисунок 1 – Список

2. Створив функцію читання (вивід вмісту на екран) даних з файлу та запис даних у файл. Код показано на рисунку.

```
Zoo create()
{
    cout << "\n\tНАЗВАННЯ: ";
    char name[50];
    cin >> name;

    cout << "\n\tКОЛИЧЕСТВО: ";
    int number;
    cin >> number;

    cout << "\n\tОКРАС: ";
    char colour[50];
    cin >> colour;

    cout << "\n\tВОЗРАСТ: ";
    int age;
    cin >> age;

    cout << "\n\tСЕМЕЙСТВО: ";
    char kind[50];
    cin >> kind;

    cout << "\n\tКОНТИНЕНТ ОБИТАННЯ: ";
    char continent[50];
    cin >> continent;

    console_clear();

    Zoo p;
    strcpy(p.kind, kind);
    strcpy(p.colour, colour);
    strcpy(p.name, name);
    strcpy(p.continent, continent);
    p.number = number;
    p.age = age;

    return p;
}
```

Рисунок 2 (а) - Функції Запису у файл

```
void form_file()
{
    cout << "\n\tКОЛ-ВО ЕЛЕМЕНТА = ";
    int count;
    cin >> count;

    FILE* file = fopen("file.dat", "wb");
    if (file == NULL)
        exit(1);

    for (int i = 0; i < count; i++)
    {
        Zoo p = create();

        fwrite(&p, sizeof(Zoo), 1, file);

        if (ferror(file))
            exit(2);
    }

    fclose(file);
}

void read_file()
{
    FILE* file = fopen("file.dat", "rb");
    Zoo p;
    cout << "НАЗВАННЯ" << setw(15) << "СЕМЕЙСТВО" << setw(30) << "КОНТИНЕНТ ОБИТАННЯ" << setw(20) << "ВОЗРАСТ" << setw(20) << "КОЛИЧЕСТВО" << setw(20) << "ОКРАС" << endl << endl;

    while (fread(&p, sizeof(Zoo), 1, file))
    {
        cout << p.name << setw(17) << p.kind << setw(27) << p.continent << setw(23) << p.age << setw(19) << p.number << setw(24) << p.colour << endl;
    }

    fclose(file);
}
```

Рисунок 2 (б) – Функції читання(вивід вмісту на екран) та запису у файл

3. Створив функції додавання та видалення файлу за вибором користувача.

Код показано на рисунку.

```
void delete_from_file(int startAge, int endAge, char* name)
{
    FILE* file = fopen("file.dat", "rb");
    FILE* tempfile = fopen("temp.dat", "wb");
    Zoo p;

    bool value = false;

    while (fread(&p, sizeof(Zoo), 1, file))
    {
        if (!(p.age >= startAge && p.age <= endAge) && (strcmp(p.name, name) == 0))
        {
            fwrite(&p, sizeof(Zoo), 1, tempfile);
            value = true;
        }
    }

    fclose(file);
    fclose(tempfile);

    if (value)
    {
        remove("file.dat");
        rename("temp.dat", "file.dat");
    }
}

void add_Zoo(Zoo t, int pos)
{
    if (pos < 1)
        cout << "\n\тНЕКОРРЕКТНЫЙ номер" << endl;
    else
    {
        FILE* file = fopen("file.dat", "rb");
        FILE* tempfile = fopen("temp.dat", "wb");
        Zoo p;
        int index = 0;

        while (fread(&p, sizeof(Zoo), 1, file))
        {
            fwrite(&p, sizeof(Zoo), 1, tempfile);
            index++;

            if (index == pos)
                fwrite(&t, sizeof(Zoo), 1, tempfile);
        }

        fclose(file);
        fclose(tempfile);

        remove("file.dat");
        rename("temp.dat", "file.dat");

        if (index < pos)
            cout << "\n\тНЕКОРРЕКТНЫЙ номер" << endl;
    }
}
```

Рисунок 3 – Функції додавання та видалення файлу

4. Також додав функції для зручності роботи з програмою. Код показано на рисунку.

```
void work_file()
{
    int oper;
    do
    {
        cout << "\n\t\t1\t\t- \tЧТЕНИЕ ФАЙЛА" << endl;
        cout << "\t\t2\t\t- \tУДАЛЕНИЕ ИЗ ФАЙЛА" << endl;
        cout << "\t\t3\t\t- \tДОБАВЛЕНИЕ В ФАЙЛ" << endl;
        cout << "\t\t4\t\t- \tОЧИСТИТЬ ФАЙЛ" << endl;
        cout << "\n\t\t0\t\t- \tНАЗАД" << endl << endl;
        cout << "\t\t\t- - >\t";

        cin >> oper;
        console_clear();
        switch (oper)
        {
            case 1:
                read_file();
                break;
            case 2:
            {
                cout << "\n\tВВЕДИТЕ НАЗВАНИЕ: ";
                char str[30];
                cin >> str;

                int start;
                cout << "\n\tВВЕДИТЕ ВОЗРАСТ: ";
                cin >> start;
                int end = start;

                delete_from_file(start, end, str);
                console_clear();
                break;
            }
            case 3:
            {
                int pos;
                cout << "\n\tПОСЛЕ КАКОГО ЭЛЕМЕНТА ДОБАВИТЬ?\t -> ";
                cin >> pos;

                Zoo p = create();
                add_Zoo(p, pos);
                break;
            }
            case 4:
            {
                cout << "\n\tВЫ ТОЧНО ХОТИТЕ ОЧИСТИТЬ СПИСОК (+ / ANYKEY)\t";
                if (_getch() == '+')
                    cout << "\n\n\tФАЙЛ ОЧИЩЕН\n", clear_file(FILENAME) == 0 ? "" : "НЕ";
                else
                    cout << "\n\n\tОТМЕНА";

                cout << "\n\n\n\tANYKEY TO CONTINUE ";
                _getch();
                console_clear();
                break;
            }
        }
    } while (oper != 0);
}
```

Рисунок 4 (а) – Функція меню роботи з програмою

6. Приклад роботи програми показано на рисунку.

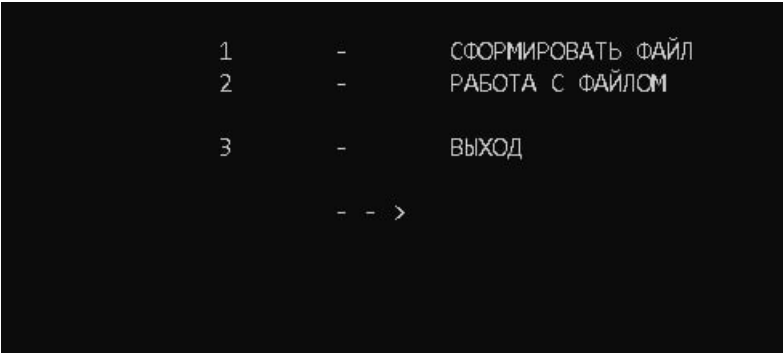


Рисунок 6 (а) – Головне меню

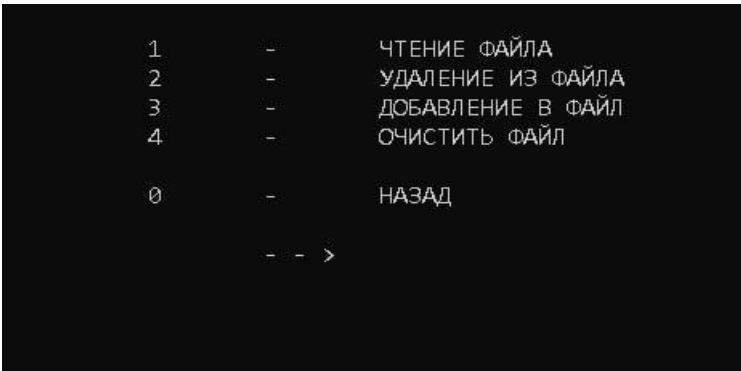


Рисунок 6 (б) – Меню роботи з файлом

НАЗВАНИЕ	СЕМЕЙСТВО	КОНТИНЕНТ ОБИТАНИЯ	ВОЗРАСТ	КОЛИЧЕСТВО	ОКРАС
Собака	собаки	Европа	8	4	белый
Кот	кошачьи	Америка	9	6	черный
мышь	грызуны	Азия	3	10	серый
	1	-	ЧТЕНИЕ ФАЙЛА		
	2	-	УДАЛЕНИЕ ИЗ ФАЙЛА		
	3	-	ДОБАВЛЕНИЕ В ФАЙЛ		
	4	-	ОЧИСТИТЬ ФАЙЛ		
	0	-	НАЗАД		
		- - >			

Рисунок 6 (в) – Читання з файлу

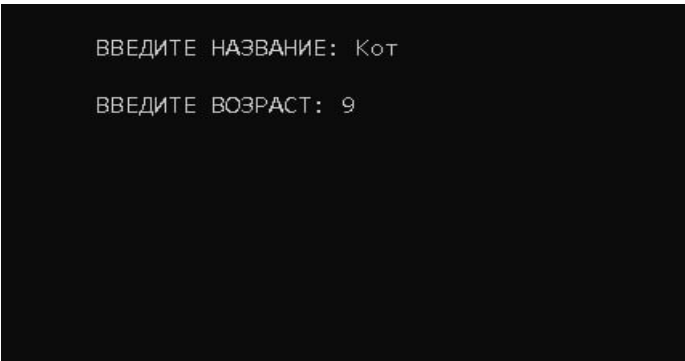


Рисунок 6 (г) – Видалення з файлу

НАЗВАНИЕ	СЕМЕЙСТВО	КОНТИНЕНТ ОБИТАНИЯ	ВОЗРАСТ	КОЛИЧЕСТВО	ОКРАС
Собака	собачих	Европа	8	4	белый
Кенгуру	сумчастые	Австралия	4	2	желтый
мышь	грызуны	Азия	3	10	серый
	1	-	ЧТЕНИЕ ФАЙЛА		
	2	-	УДАЛЕНИЕ ИЗ ФАЙЛА		
	3	-	ДОБАВЛЕНИЕ В ФАЙЛ		
	4	-	ОЧИСТИТЬ ФАЙЛ		
	0	-	НАЗАД		
	-	-	>		

Рисунок 6 (д) – Список після видалення елементу

ПОСЛЕ КАКОГО ЭЛЕМЕНТА ДОБАВИТЬ? -> 2

НАЗВАНИЕ: Страус

КОЛИЧЕСТВО: 5

ОКРАС: черный

ВОЗРАСТ: 2

СЕМЕЙСТВО: пернатых

КОНТИНЕНТ ОБИТАНИЯ: Австралия

Рисунок 6 (е) – Додавання елементу

НАЗВАНИЕ	СЕМЕЙСТВО	КОНТИНЕНТ ОБИТАНИЯ	ВОЗРАСТ	КОЛИЧЕСТВО	ОКРАС
Собака	собачих	Европа	8	4	белый
Кенгуру	сумчастые	Австралия	4	2	желтый
Страус	пернатых	Австралия	2	5	черный
мышь	грызуны	Азия	3	10	серый
	1	-	ЧТЕНИЕ ФАЙЛА		
	2	-	УДАЛЕНИЕ ИЗ ФАЙЛА		
	3	-	ДОБАВЛЕНИЕ В ФАЙЛ		
	4	-	ОЧИСТИТЬ ФАЙЛ		
	0	-	НАЗАД		
	-	-	>		

Рисунок 6 (з) – Список після додавання елементу

Висновок: Освоїв тему «Динамічні списки». Навчився працювати з динамічними списками.