

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) Додали динамічний список прикладної області. Код показано на рисунку.

```

1  #define FILENAME "file.dat"
2  #define _CRT_SECURE_NO_WARNINGS
3  #include <iostream>
4  #include <iomanip>
5  #include <conio.h>
6  #include <Windows.h>
7  using namespace std;
8
9  struct PoultryFarm
10 {
11     char name[50];
12     int number;
13     char colour[50];
14     char kind[50];
15     int age;
16 };
17
18 void console_clear();
19 void form_file();
20 void read_file();
21 void work_file();
22 void delete_from_file(int startAge, int endAge, char* name);
23 int clear_file(const char* filename);
24 PoultryFarm create();
    
```

Рисунок 1 – Список

- 2) Додали функцію читання (вивід вмісту на екран) даних з файлу та запис даних у файл.

```

57 PoultryFarm create()
58 {
59     cout << "Название: ";
60     char name[50];
61     cin >> name;
62
63     cout << "Количество: ";
64     int number;
65     cin >> number;
66
67     cout << "Окрас: ";
68     char colour[50];
69     cin >> colour;
70
71     cout << "Возраст: ";
72     int age;
73     cin >> age;
74
75     cout << "Семейство: ";
76     char kind[50];
77     cin >> kind;
78
79     console_clear();
80
81     PoultryFarm p;
82     strcpy(p.kind, kind);
83     strcpy(p.colour, colour);
84     strcpy(p.name, name);
85     p.number = number;
86     p.age = age;
87
88     return p;
89 }

```

Рисунок 2.1 – Функції Запису у файл

```

91 void form_file()
92 {
93     cout << "Количество элементов: ";
94     int count;
95     cin >> count;
96
97     FILE* file = fopen("file.dat", "wb");
98     if (file == NULL)
99         exit(1);
100
101     for (int i = 0; i < count; i++)
102     {
103         PoultryFarm p = create();
104
105         fwrite(&p, sizeof(PoultryFarm), 1, file);
106
107         if (ferror(file))
108             exit(2);
109     }
110
111     fclose(file);
112 }
113
114 void read_file()
115 {
116     FILE* file = fopen("file.dat", "rb");
117     PoultryFarm p;
118     cout << "Название" << setw(15) << "Семейство" << setw(30) << "Возраст" << setw(20) << "Количество" << setw(20) << "Окрас" << endl << endl;
119
120     while (fread(&p, sizeof(PoultryFarm), 1, file))
121     {
122         cout << p.name << setw(17) << p.kind << setw(27) << p.age << setw(19) << p.number << setw(24) << p.colour << endl;
123     }
124
125     fclose(file);
126 }

```

2.2 – Функції читання(вивід вмісту на екран) та запису у файл

3) Додали функції додавання та видалення файлу за вибором користувача.

Код показано на рисунку.

```

139 void delete_from_file(int startAge, int endAge, char* name)
140 {
141     FILE* file = fopen("file.dat", "rb");
142     FILE* tempfile = fopen("temp.dat", "wb");
143     PoultryFarm p;
144
145     bool value = false;
146
147     while (fread(&p, sizeof(PoultryFarm), 1, file))
148     {
149         if (!((p.age >= startAge && p.age <= endAge) && (strcmp(p.name, name) == 0)))
150         {
151             fwrite(&p, sizeof(PoultryFarm), 1, tempfile);
152             value = true;
153         }
154     }
155
156     fclose(file);
157     fclose(tempfile);
158
159     if (value)
160     {
161         remove("file.dat");
162         rename("temp.dat", "file.dat");
163     }
164 }
165

```

Рисунок 3.1 – Перша частина

```

166 void add_PoultryFarm(PoultryFarm t, int pos)
167 {
168     if (pos < 1)
169         cout << "\n\tНЕКОРРЕКТНЫЙ НОМЕР" << endl;
170     else
171     {
172         FILE* file = fopen("file.dat", "rb");
173         FILE* tempfile = fopen("temp.dat", "wb");
174         PoultryFarm p;
175         int index = 0;
176
177         while (fread(&p, sizeof(PoultryFarm), 1, file))
178         {
179             fwrite(&p, sizeof(PoultryFarm), 1, tempfile);
180             index++;
181
182             if (index == pos)
183                 fwrite(&t, sizeof(PoultryFarm), 1, tempfile);
184         }
185
186         fclose(file);
187         fclose(tempfile);
188
189         remove("file.dat");
190         rename("temp.dat", "file.dat");
191
192         if (index < pos)
193             cout << "\n\tНЕКОРРЕКТНЫЙ НОМЕР" << endl;
194     }
195 }
196

```

Рисунок 3.2 – Друга частина. Функції додавання та видалення файлу

4) Також додали функції для зручності роботи з програмою.

```
197 void work_file()
198 {
199     int oper;
200     do
201     {
202         cout << "1. Просмотр файла" << endl;
203         cout << "2. Удаление из файла" << endl;
204         cout << "3. Добавление в файл" << endl;
205         cout << "4. Очистить файл" << endl;
206         cout << "0. Назад" << endl << endl;
207         cout << "Выберите вариант: ";
208
209         cin >> oper;
210         console_clear();
211         switch (oper)
212         {
213             case 1:
214                 read_file();
215                 break;
216             case 2:
217             {
218                 cout << "Название: ";
219                 char str[30];
220                 cin >> str;
221
222                 int start;
223                 cout << "Начиная с: ";
224                 cin >> start;
225                 int end = start;
226
227                 delete_from_file(start, end, str);
228                 console_clear();
229                 break;
230             }
231             case 3:
232             {
233                 int pos;
234                 cout << "После какого элемента добавить? -> ";
235                 cin >> pos;
236
237                 PoultryFarm p = create();
238                 add_PoultryFarm(p, pos);
239                 break;
240             }
241             case 4:
242             {
243                 cout << "Вы точно хотите очистить список? (+ / ANYKEY)\n";
244                 if (_getch() == '+')
245                     cout << "Файл очищен\n", clear_file(FILENAME) == 0 ? "" : "НЕ";
246                 else
247                     cout << "Отмена";
248
249                 cout << "\nANYKEY TO CONTINUE ";
250                 _getch();
251                 console_clear();
252                 break;
253             }
254         }
255     } while (oper != 0);
256 }
```

Рисунок 4.1 – Функція меню роботи з програмою

```
128 int clear_file(const char* filename)
129 {
130     FILE* f = NULL;
131
132     if (fopen_s(&f, filename, "wb") != 0)
133         return -1;
134
135     fclose(f);
136     return 0;
137 }
```

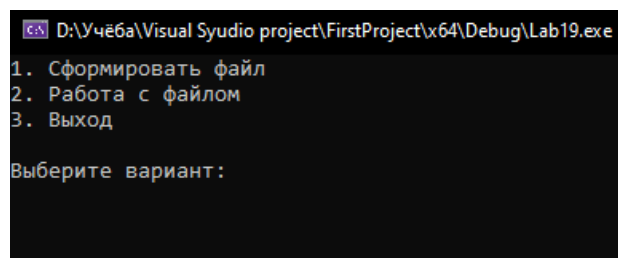
Рисунок 4.2 – Функція очищення списку

5) Створили функцію main з головним меню програми.

```
26 int main()
27 {
28     SetConsoleOutputCP(1251);
29     SetConsoleCP(1251);
30
31     int oper;
32
33     do
34     {
35         cout << "1. Сформировать файл" << endl;
36         cout << "2. Работа с файлом" << endl;
37         cout << "3. Выход" << endl << endl;
38         cout << "Выберите вариант: ";
39
40         cin >> oper;
41         console_clear();
42         switch (oper)
43         {
44             case 1:
45                 form_file();
46                 break;
47             case 2:
48                 work_file();
49                 break;
50             default:
51                 break;
52         }
53     } while (oper != 3);
54 }
55
```

Рисунок 5 – Функція main

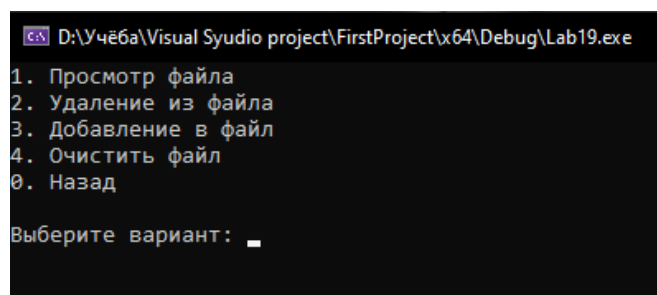
6) Приклад роботи програми.



```
D:\Учёба\Visual Syudio project\FirstProject\x64\Debug\Lab19.exe
1. Сформировать файл
2. Работа с файлом
3. Выход

Выберите вариант:
```

Рисунок 6.1 – Головне меню



```
D:\Учёба\Visual Syudio project\FirstProject\x64\Debug\Lab19.exe
1. Просмотр файла
2. Удаление из файла
3. Добавление в файл
4. Очистить файл
0. Назад

Выберите вариант: _
```

Рисунок 6.2 – Меню роботи з файлом

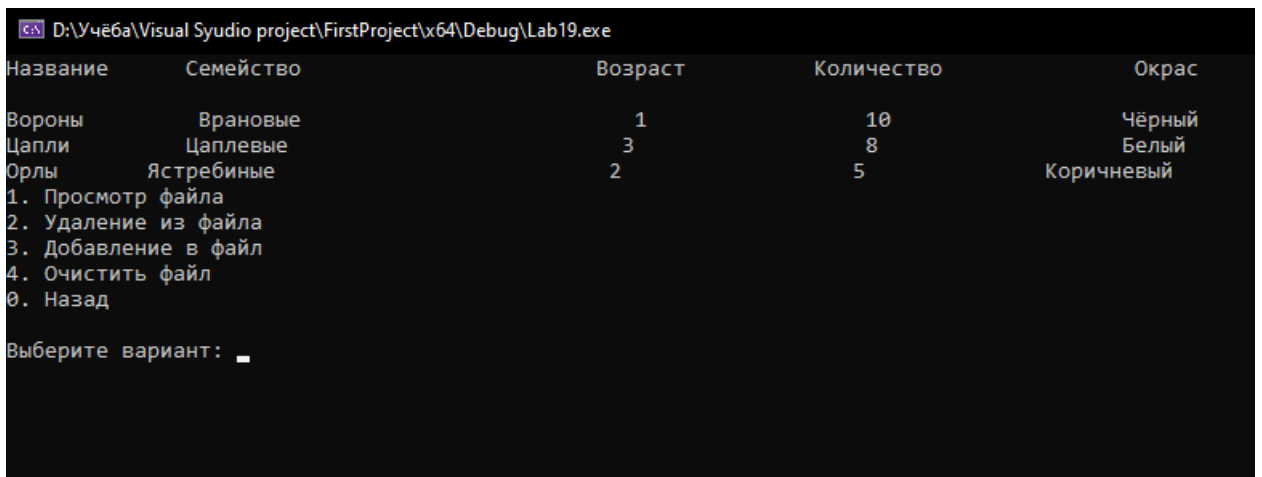


Рисунок 6.3 – Читання з файлу

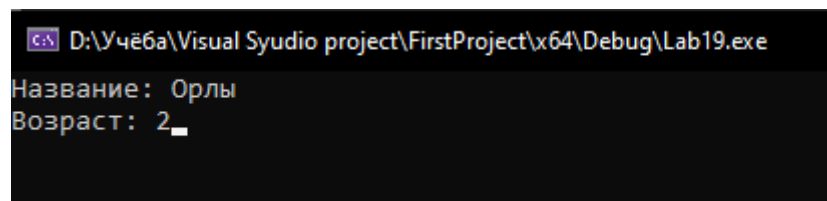


Рисунок 6.4 – Видалення з файлу

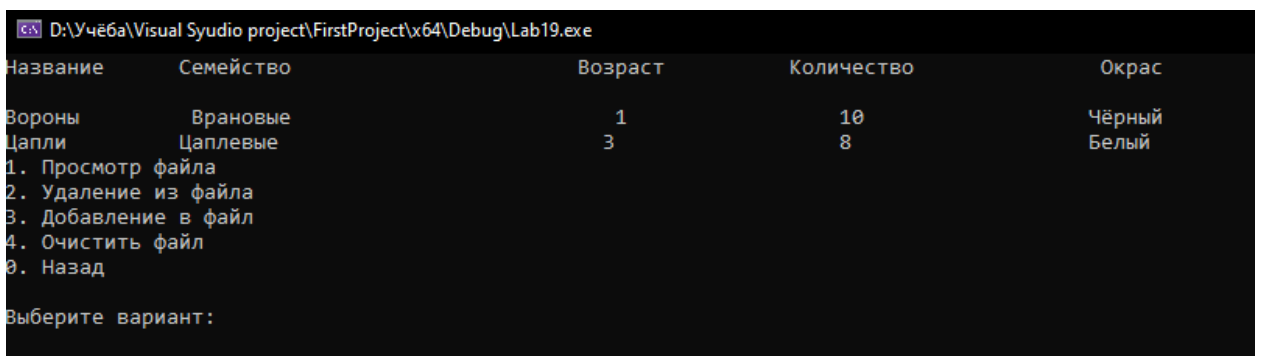


Рисунок 6.5 – Список після видалення елементу

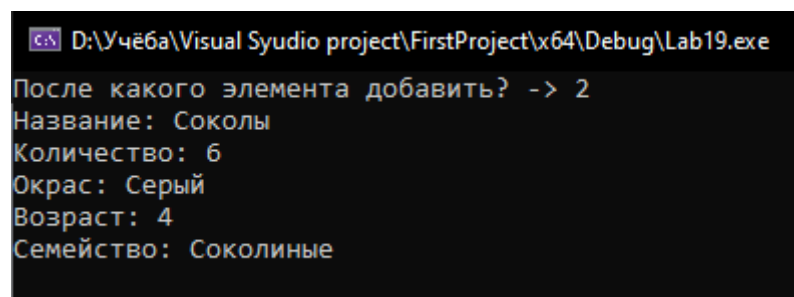


Рисунок 6.6 – Додавання елементу

D:\Учеба\Visual Syudio project\FirstProject\x64\Debug\Lab19.exe

Название	Семейство	Возраст	Количество	Окрас
Вороны	Врановые	1	10	Чёрный
Цапли	Цаплевые	3	8	Белый
Соколы	Соколиные	4	6	Серый
1. Просмотр файла				
2. Удаление из файла				
3. Добавление в файл				
4. Очистить файл				
0. Назад				
Выберите вариант: <input type="text"/>				

Рисунок 6.7 – Список після додавання елементу

Висновок: Освоїли тему «Динамічні списки». Навчилися працювати з динамічними списками.

github: https://github.com/Nikita-Stetsenko/lab_v2.0