

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Інститут комп'ютерних наук та інформаційних технологій

Кафедра систем штучного інтелекту



Лабораторна робота №8

На тему: **«Блоковий ввід-вивід»**

Виконав:

ст. гр. КН – 109

Паберівський Роман

Прийняв:

Гасько Р. Т.

Львів – 2018

Лабораторна робота №8

Тема роботи: Блоковий ввід-ввід.

Мета роботи: робота з двійковими файлами, організація вводу-виводу структурованої інформації і її зберігання на зовнішніх носіях.

Постановка завдання

Сформувати двійковий файл із елементів, заданої у варіанті структури, роздрукувати його вміст, виконати знищення і додавання елементів у відповідності зі своїм варіантом, використовуючи для пошуку елементів, що знищуються чи додаються, функцію. Формування, друк, додавання й знищення елементів оформити у вигляді функції. Передбачити повідомлення про помилки при відкритті файлу й виконанні операцій вводу/виводу.

Варіант 20

Структура «Держава»:

- Назва;
- Державна мова;
- Грошова одиниця;
- Курс валюти відносно €;

Знищити елемент із зазначеною назвою, додати 2 елементи в кінець файлу.

Код

```
1 #include <stdio.h>
2 #include <string.h>
3 #include <stdlib.h>
4 #include <ctype.h>
5 #define STR_LEN 31
6 #define SIZE 100
7
8 int size = 0;
9
10 typedef struct
11 {
12     char name[STR_LEN];
13     char language[STR_LEN];
14     char currency[STR_LEN];
15     double exchange_rate;
16 } COUNTRY;
17
18 int check_string(char* s)
19 {
20     int isCorrect = 1;
21     for (int i = 0; i < strlen(s) - 2; i++)
22         if (!(isalpha(s[i]) || s[i] == ' ' || s[i] == ','))
23         {
24             isCorrect = 0;
25             break;
26         }
```

```
26     }
27     return isCorrect;
28 }
29
30 int check_number(char* s)
31 {
32     int isCorrect = 1;
33     for (int i = 0; i < strlen(s) - 2; i++)
34         if (!(isdigit(s[i]) || s[i] == '.' || s[i] == '-'))
35         {
36             isCorrect = 0;
37             break;
38         }
39     return isCorrect;
40 }
41
42 int readFromFile(COUNTRY *c, char* fname)
43 {
44     FILE *f = fopen(fname, "rb");
45     if (!f) {
46         printf("Can't open file \"%s\"!\n", fname);
47         return 1;
48     }
```

```

48 }
49 char *estr;
50 for(int i = 0; 1; i++){
51     int add = 1;
52     char name[STR_LEN];
53     estr = fgets(name, STR_LEN, f);
54     if (!estr && i == 0) {
55         printf("file \"%s\" is empty!\n", fname);
56         break;
57     }
58     if (!check_string(name)) {
59         printf("Error: row %d. Name must consist only letters!\n", i*4 + 1);
60         add = 0;
61     }
62     if (!estr) {
63         printf("Not enough information about %d country!\n", size + 1);
64         break;
65     }
66
67     char language[STR_LEN];
68     estr = fgets(language, STR_LEN, f);
69     if (!check_string(language)) {
70         printf("Error: row %d. Language must consist only letters!\n", i * 4 + 2);

```

```

71         add = 0;
72     }
73     if (!estr) {
74         printf("Not enough information about %d country!\n", size + 1);
75         break;
76     }
77
78     char currency[STR_LEN];
79     estr = fgets(currency, STR_LEN, f);
80     if (!check_string(currency)) {
81         printf("Error: row %d. Currency must consist only letters!\n", i * 4 + 3);
82         add = 0;
83     }
84     if (!estr) {
85         printf("Not enough information about %d country!\n", size + 1);
86         break;
87     }
88
89     char rate[STR_LEN];
90     estr = fgets(rate, STR_LEN, f);
91     if (!check_number(rate)) {
92         printf("Error: row %d. Exchange rate must be floating value!\n", i * 4 + 3);
93         add = 0;

```

```

94     }
95     double exchange_rate = atof(rate);
96     if (exchange_rate <= 0)
97     {
98         printf("Error: row %d. Exchange rate must be >= 0!\n", i * 4 + 3);
99         add = 0;
100    }
101    if (add)
102    {
103        strcpy(c[size].name, name);
104        c[size].name[strlen(name) - 2] = '\0';
105        strcpy(c[size].language, language);
106        c[size].language[strlen(language) - 2] = '\0';
107        strcpy(c[size].currency, currency);
108        c[size].currency[strlen(currency) - 2] = '\0';
109        c[size].exchange_rate = exchange_rate;
110        size++;
111    }
112    if(!estr) break;
113 }
114 fclose(f);
115 return size > 0 ? 0 : 1;

```

```

116 }
117
118 void print(COUNTRY c[])
119 {
120     for (int i = 0; i < (STR_LEN + 1) * 3 + 16; i++) printf(" ");
121     printf("\n| # | %-30s| %-30s| %-30s| %-7s|\n", "Name", "Language", "Currency", "Rate");
122     for (int i = 0; i < (STR_LEN + 1) * 3 + 16; i++) printf("-");
123     for (int i = 0; i < size; i++)
124     {
125         printf("\n| %-4d| %-30s| %-30s| %-30s| %-7.3lf|\n", i+1, c[i].name, c[i].language, c[i].currency, c[i].exchange_rate);
126         for (int i = 0; i < (STR_LEN + 1) * 3 + 16; i++) printf("-");
127     }
128 }
129
130 int searchByName(COUNTRY c[], char* name)
131 {
132     for (int i = 0; i < size; i++)
133         if (strcmp(name, c[i].name) == 0)
134             return i;
135     return size;
136 }
137
138 void removeElement(COUNTRY c[], int ind)

```

```

139 {
140     for (int i = ind; i < size - 1; i++)
141         c[i] = c[i + 1];
142     size--;
143 }
144
145 void addCountry(COUNTRY c[])
146 {
147     char str[STR_LEN];
148     printf("Enter name: ");
149     while(1)
150     {
151         fgets(str, STR_LEN, stdin);
152         str[strlen(str) - 1] = '\0';
153         if (!check_string(str))
154             printf("Error: Name must consist only letters!\nTry again: ");
155         else break;
156     }
157     strcpy(c[size].name, str);
158     printf("Enter language: ");
159     while (1)
160     {
161         fgets(str, STR_LEN, stdin);

```

```

162     str[strlen(str) - 1] = '\0';
163     if (!check_string(str))
164         printf("Error: Language must consist only letters!\nTry again: ");
165     else break;
166 }
167 strcpy(c[size].language, str);
168
169 printf("Enter Currency: ");
170 while (1)
171 {
172     fgets(str, STR_LEN, stdin);
173     str[strlen(str) - 1] = '\0';
174     if (!check_string(str))
175         printf("Error: Currency must consist only letters!\nTry again: ");
176     else break;
177 }
178 strcpy(c[size].currency, str);
179
180 printf("Enter Exchange rate: ");
181 while (1)
182 {
183     fgets(str, STR_LEN, stdin);

```

```

184     str[strlen(str) - 1] = '\0';
185     if (!check_number(str) || atof(str) <= 0)
186         printf("Error: Exchange rate must be floating value and must be >= 0!\nTry again: ");
187     else break;
188 }
189 c[size].exchange_rate = atof(str);
190 size++;
191 }
192
193 void SaveToFile(COUNTRY *c, char* fname)
194 {
195     FILE *fout = fopen(fname, "wb");
196     for(int i = 0; i < size; i++)
197     {
198         fprintf(fout, "%s\n", c[i].name);
199         fprintf(fout, "%s\n", c[i].language);
200         fprintf(fout, "%s\n", c[i].currency);
201         fprintf(fout, "%lf", c[i].exchange_rate);
202         if(i != size-1) fprintf(fout, "\n");
203     }
204     fclose(fout);
205 }
206
207 int main(void)

```

```
207 int main(void)
208 {
209     COUNTRY countries[SIZE];
210     char fname[STR_LEN];
211     printf("Enter name of file with data: ");
212     scanf("%s", fname);
213     getchar();
214     if (readFromFile(countries, fname) == 0)
215     {
216         print(countries);
217         printf("\n\n");
218         printf("Enter name of country, witch must be deleted: ");
219         char name[STR_LEN];
220         fgets(name, STR_LEN, stdin);
221         name[strlen(name) - 1] = '\0';
222         int index = searchByName(countries, name);
223         if (index >= size) printf("Not found!\n");
224         else
225         {
226             removeElement(countries, index);
227             print(countries);
228             printf("\n\n");
229         }
230         for (int i = 0; i < 2; i++)
```

```
231     {
232         printf("Enter data about %d country:\n", i + 1);
233         addCountry(countries);
234         printf("\n");
235     }
236     print(countries);
237     printf("\n\n");
238 }
239 SaveToFile(countries, fname);
240
241 getchar();
242 return 0;
243 }
```

Результат виконання програми

```
jharvard@appliance (~/lab): ./laba_8
Enter name of file with data: Data.bin
Not enough information about 6 country!
```

#	Name	Language	Currency	Rate
1	Ukrai	Ukraini	Ukrainian hryvn	0.035
2	United States of Ameri	Engli	United States doll	1.000
3	Austral	Engli	Australian doll	1.250
4	Franc	Francia	Eur	1.200
5	Canad	Englis	Canadian dolla	0.800

```
Enter name of country, witch must be deleted: Franc
```

#	Name	Language	Currency	Rate
1	Ukrai	Ukraini	Ukrainian hryvn	0.035

Terminal

2	United States of Ameri	Engli	United States doll	1.000
3	Austral	Engli	Australian doll	1.250
4	Canad	Englis	Canadian dolla	0.800

```
Enter data about 1 country:
Enter name: 
```

Terminal