**Правительство Российской Федерации**

**Федеральное государственное автономное образовательное учреждение высшего профессионального образования   
"Национальный исследовательский университет   
"Высшая школа экономики"**

Московский институт электроники и математики Национального

исследовательского университета "Высшая школа экономики"

Департамент прикладной математики

**ОТЧЕТ**

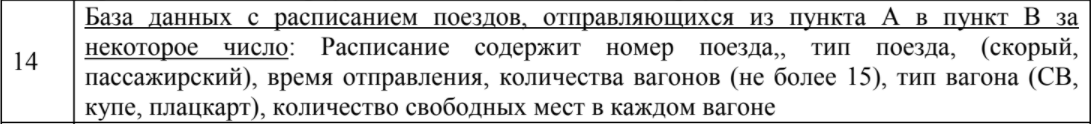
**По лабораторной работе № 11**

**По курсу «Алгоритмизация и программирование»**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | |  | ФИО студента | | Номер группы | Дата |
| Колодин Матвей Алексеевич | БПМ213 | 10.06.22 |
|  |
|  |
|  |

**Москва – 2022 г.**

**ЗАДАНИЕ (вариант № 14)**

****

**РЕШЕНИЕ**

**Source.c**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <locale.h>

#include <string.h>

#include <io.h>

#include "trains.h"

#define Tdata train

int menu(char \*name)

{

int variant = 1;

while (1)

{

printf("Menu:\n");

printf("%d - %s\n", 0, "Close application");

printf("%d - %s\n", 1, "Print all records");

printf("%d - %s\n", 2, "Push a record");

printf("%d - %s\n", 3, "Delete certain record");

printf("%d - %s\n", 4, "Search a record");

printf("%d - %s\n", 5, "Edit record");

printf("%d - %s\n", 6, "Sorting records");

printf("%d - %s\n", 7, "Create a database");

printf("%d - %s\n", 8, "Print one record");

printf("Choose an option:\n");

scanf\_s("%d", &variant);

getchar();

switch (variant) {

case 0:

return 0;

case 1:

bd\_all\_write(name);

break;

case 2:

bd\_add\_item(name);

break;

case 3:

bd\_delete(name);

break;

case 4:

bd\_search(name);

break;

case 5:

bd\_edit(name);

break;

case 6:

bd\_sort(name);

break;

case 7:

bd\_create(name);

break;

case 8:

bd\_write\_item\_and\_titles(name);

break;

}

}

return 0;

}

void search\_option\_menu()

{

puts("");

printf("%d - %s\n", 0, "Close menu");

printf("%d - %s\n", 1, "ID of the train");

printf("%d - %s\n", 2, "Train type");

printf("%d - %s\n", 3, "Departure time");

printf("%d - %s\n", 4, "Wagon quantity");

printf("%d - %s\n", 5, "Certain type of wagon availability");

printf("%d - %s\n", 6, "Available tickets in one wagon");

}

void sort\_option\_menu()

{

puts("");

printf("%d - %s\n", 0, "Close menu");

printf("%d - %s\n", 1, "ID of the train");

printf("%d - %s\n", 2, "Train type");

printf("%d - %s\n", 3, "Departure time");

printf("%d - %s\n", 4, "Wagon quantity");

}

**search\_sort.c**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <locale.h>

#include <string.h>

#include "trains.h"

#define Tdata train

int bd\_compare(char\* val\_str, int val\_int, Tdata temp, int v)

{

int leng = 1, checker;

unsigned int i;

if (v == 1)

{

if (val\_int != temp.id)

{

return 0;

}

}

if (v == 2)

{

if (val\_int != temp.train\_type)

{

return 0;

}

}

if (v == 3)

{

leng = (int)strlen(val\_str);

if (strncmp(val\_str, temp.time, leng) != 0)

{

return 0;

}

}

if (v == 4)

{

if (val\_int != temp.wagon\_q)

{

return 0;

}

}

if (v == 5)

{

checker = 0;

for (i = 0; i < temp.wagon\_q; i++){

if (val\_int == temp.wagons[i].wagon\_type){

checker = 1;

}

if (checker == 0){

return 0;

}

}

}

if (v == 6)

{

checker = 1;

for (i = 0; i < temp.wagon\_q; i++){

if (val\_int <= temp.wagons[i].tickets){

checker = 0;

}

if (checker){

return 0;

}

}

}

return 1;

}

void bd\_item\_search(int n, char\* name){

FILE\* fout;

Tdata temp;

char val\_str[15];

int k = 0, val\_int;

if ((fout = fopen(name, "rb")) == NULL)

{

puts("Error while opening file.\n");

}

else

{

printf("Enter the data to specify the search.\n");

switch (n)

{

case 1:

printf("Enter ID: ");

scanf("%u%\*c", &val\_int);

break;

case 2:

printf("Enter train type\nWhere 0 - Express, 1 - Passenger: ");

scanf("%u%\*c", &val\_int);

break;

case 3:

printf("Enter departure time: ");

fgets(val\_str, 10, stdin);

val\_str[strlen(val\_str) - 1] = 0;

break;

case 4:

printf("Enter wagon quantity: ");

scanf("%u%\*c", &val\_int);

break;

case 5:

printf("Enter wagon type\nWhere 0 - CB, 1 - Compartment, 2 - reserved seat: ");

scanf("%u%\*c", &val\_int);

break;

case 6:

printf("Enter quantity of available tickets:");

scanf("%u%\*c", &val\_int);

break;

}

bd\_print\_titles();

while (!feof(fout))

{

fread(&temp, sizeof(Tdata), 1, fout);

if (!feof(fout))

{

if (bd\_compare(val\_str, val\_int, temp, n))

{

write\_single\_item(temp);

k++;

}

}

}

puts("");

}

fclose(fout);

if (k == 0)

{

printf("No matches.\n\n");

}

}

void bd\_search(char\* name)

{

int n = 0;

search\_option\_menu();

printf("Choose an option: ");

scanf("%d%\*c", &n);

puts("");

if (n != 0)

{

bd\_item\_search(n, name);

}

}

void bd\_sort(char\* name)

{

FILE\* fin = fopen(name, "r+b");

Tdata tra1, tra2, temp;

unsigned int curr = 0;

int n, file\_size, pos, option = 0;

if (fin == NULL)

{

puts("Error while opening file.");

}

else

{

fseek(fin, 0, SEEK\_END);

file\_size = (int)ftell(fin);

n = file\_size / sizeof(Tdata);

printf("All sorting comaparators:\n");

sort\_option\_menu();

printf("Choose an option: ");

scanf("%d", &option);

switch (option) {

case 1:

for (int i = 0; i < n; i++) {

for (int j = 0; j < n - i - 1; j++) {

fseek(fin, j \* sizeof(Tdata), SEEK\_SET);

fread(&tra1, sizeof(Tdata), 1, fin);

fread(&tra2, sizeof(Tdata), 1, fin);

if (!feof(fin)) {

if (tra1.id > tra2.id) {

temp = tra1;

tra1 = tra2;

tra2 = temp;

fseek(fin, j \* sizeof(Tdata), SEEK\_SET);

fwrite(&tra1, sizeof(Tdata), 1, fin);

fwrite(&tra2, sizeof(Tdata), 1, fin);

}

}

}

}

break;

case 2:

for (int i = 0; i < n; i++) {

for (int j = 0; j < n - i - 1; j++) {

fseek(fin, j \* sizeof(Tdata), SEEK\_SET);

fread(&tra1, sizeof(Tdata), 1, fin);

fread(&tra2, sizeof(Tdata), 1, fin);

if (!feof(fin)) {

if (tra1.train\_type > tra2.train\_type) {

temp = tra1;

tra1 = tra2;

tra2 = temp;

fseek(fin, j \* sizeof(Tdata), SEEK\_SET);

fwrite(&tra1, sizeof(Tdata), 1, fin);

fwrite(&tra2, sizeof(Tdata), 1, fin);

}

}

}

}

break;

case 3:

for (int i = 0; i < n; i++) {

for (int j = 0; j < n - i - 1; j++) {

fseek(fin, j \* sizeof(Tdata), SEEK\_SET);

fread(&tra1, sizeof(Tdata), 1, fin);

fread(&tra2, sizeof(Tdata), 1, fin);

if (!feof(fin)) {

pos = strcmp(tra1.time, tra2.time);

if (pos > 0) {

temp = tra1;

tra1 = tra2;

tra2 = temp;

fseek(fin, j \* sizeof(Tdata), SEEK\_SET);

fwrite(&tra1, sizeof(Tdata), 1, fin);

fwrite(&tra2, sizeof(Tdata), 1, fin);

}

}

}

}

break;

case 4:

for (int i = 0; i < n; i++) {

for (int j = 0; j < n - i - 1; j++) {

fseek(fin, j \* sizeof(Tdata), SEEK\_SET);

fread(&tra1, sizeof(Tdata), 1, fin);

fread(&tra2, sizeof(Tdata), 1, fin);

if (!feof(fin)) {

if (tra1.wagon\_q > tra2.wagon\_q) {

temp = tra1;

tra1 = tra2;

tra2 = temp;

fseek(fin, j \* sizeof(Tdata), SEEK\_SET);

fwrite(&tra1, sizeof(Tdata), 1, fin);

fwrite(&tra2, sizeof(Tdata), 1, fin);

}

}

}

}

}

printf("\n");

for (int i = 0; i < n; i++)

{

fseek(fin, i \* sizeof(Tdata), SEEK\_SET);

curr = i + 1;

fwrite(&curr, sizeof(curr), 1, fin);

}

}

fclose(fin);

}

**delete\_edit.c**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <locale.h>

#include <string.h>

#include "trains.h"

#define Tdata train

void bd\_delete(char\* file\_name) {

int num;

FILE\* fin;

train tra;

unsigned int curr = 0;

int n;

printf("Enter record number: ");

scanf\_s("%d", &num);

if ((fin = fopen(file\_name, "r+b")) != NULL) {

while (!feof(fin)) {

fread(&tra, sizeof(tra), 1, fin);

if (!feof(fin)) {

if (num == tra.number) {

break;

}

}

}

while (!feof(fin)) {

fread(&tra, sizeof(tra), 1, fin);

if (feof(fin)) break;

fseek(fin, (-2)\*(int)sizeof(tra), SEEK\_CUR);

fwrite(&tra, sizeof(tra), 1, fin);

fseek(fin, sizeof(tra), SEEK\_CUR);

}

chsize(\_fileno(fin), ftell(fin) - sizeof(tra));

n = (int)ftell(fin) / sizeof(Tdata);

for (int i = 0; i < n; i++)

{

fseek(fin, i \* sizeof(Tdata), SEEK\_SET);

curr = i + 1;

fwrite(&curr, sizeof(curr), 1, fin);

}

fclose(fin);

}

else

printf("Error while deleting");

}

void bd\_edit(char\* file\_name) {

int id, c, num;

FILE\* fin;

train tra;

printf("Enter the number of the train: ");

scanf\_s("%d", &id);

if ((fin = fopen(file\_name, "r+b")) != NULL) {

while (!feof(fin)) {

fread(&tra, sizeof(tra), 1, fin);

if (!feof(fin)) {

if (tra.number == id) {

break;

}

}

}

if (feof(fin)) {

printf("There is no such train.\n");

return;

}

printf("Choose what you want to edit:\n");

printf("0 for number of the train\n"

"1 for train type\n"

"2 for departure time\n"

"3 for quantity of wagons\n"

"4 for wagon info\n");

scanf\_s("%d", &c);

if (c == 4){

printf("Enter number of wagon: ");

scanf\_s("%d", &num);

printf("5 for wagon type\n"

"6 for available tickets quantity\n");

scanf\_s("%d", &c);

}

unsigned int val;

char value[10];

printf("Enter new info:\n");

scanf("%\*c");

switch (c) {

case 0:

scanf\_s("%u", &val);

tra.number = val;

break;

case 1:

printf("Where 0 - express, 1 - passenger: ");

scanf\_s("%u", &val);

tra.train\_type = val;

break;

case 2:

fgets(tra.time, 10, stdin);

tra.time[strlen(tra.time) - 1] = '\0';

break;

case 3:

scanf\_s("%u", &val);

tra.wagon\_q = val;

break;

case 5:

printf("Where 0 - CB, 1 - compartment, 2 - reserved seat: ");

scanf\_s("%u", &val);

tra.wagons[num - 1].wagon\_type = val;

break;

case 6:

scanf\_s("%u", &val);

tra.wagons[num - 1].tickets = val;

break;

}

fseek(fin, -1 \* (int)sizeof(tra), SEEK\_CUR);

fwrite(&tra, sizeof(tra), 1, fin);

fclose(fin);

}

else

printf("Error while editing");

}

**main.c**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <locale.h>

#include <string.h>

#include <io.h>

#include "trains.h"

#define Tdata train

int main()

{

char name[] = "db";

menu(name);

return 0;

}

**print\_add.c**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h> //подключаем основную библиотеку

#include <stdlib.h>

#include <math.h> //подключаем библиотеку математики

#include <locale.h> //подключаем библиотеку языков

#include <string.h>

#include "trains.h"

#define Tdata train

void write\_single\_item(Tdata tr)

{

int c = 0;

printf("%-8u", tr.number);

printf(" | ");

printf("%-8u", tr.id);

printf(" | ");

if (tr.train\_type){

printf("%-15s", "Passenger");

}

else{

printf("%-15s", "Express");

}

printf(" | ");

printf("%-15s", tr.time);

printf(" | ");

printf("%-15u", tr.wagon\_q);

printf(" | ");

for (unsigned int i = 0; i < tr.wagon\_q; i++)

{

if (c % 2 == 0 && c!=0){

printf("\n");

printf("%-73s", "");

printf(" | ");

}

printf("%-12u", tr.wagons[i].w\_number);

printf(" | ");

if (tr.wagons[i].wagon\_type == 0){

printf("%-15s", "CB");

}

else if (tr.wagons[i].wagon\_type == 1){

printf("%-15s", "Compartment");

}

else if (tr.wagons[i].wagon\_type == 2){

printf("%-15s", "Reserved seat");

}

printf(" | ");

printf("%-12u", tr.wagons[i].tickets);

printf(" | ");

c++;

}

printf("\n");

for (int i = 0; i < 165; i++)

{

printf("-");

}

printf("\n");

}

Tdata new\_item(int c)

{

Tdata tr;

char temp[50];

tr.number = c;

c++;

printf("Enter train ID: ");

scanf\_s("%u", &tr.id);

printf("Where 0 - express, 1 - passenger\nEnter train type: ");

scanf\_s("%u%\*c", &tr.train\_type);

printf("Enter train departure time: ");

fgets(tr.time, 10, stdin);

tr.time[strlen(tr.time) - 1] = '\0';

printf("Enter wagon quantity: ");

scanf\_s("%u", &tr.wagon\_q);

for (unsigned int i = 0; i < tr.wagon\_q; i++){

printf("Wagon #%d:\n", i);

tr.wagons[i].w\_number = i + 1;

printf("Where 0 - CB, 1 - compartment, 2 - reserved seat\nEnter wagon type: ");

scanf\_s("%u", &tr.wagons[i].wagon\_type);

printf("Enter number of available tickets: ");

scanf\_s("%u", &tr.wagons[i].tickets);

}

printf("\n\n");

return tr;

}

int bd\_create(char \*name)

{

int n;

Tdata bk;

FILE\* fin = fopen(name, "w+b");

if (fin == NULL)

{

printf("Error while opening file.");

fclose(fin);

return 1;

}

else

{

puts("Enter train quantity:");

printf("n = ");

scanf\_s("%d%\*c", &n);

puts("Enter the data:");

for (int i = 0; i < n; i++)

{

bk = new\_item(i + 1);

fwrite(&bk, sizeof(Tdata), 1, fin);

}

fclose(fin);

}

return 0;

}

void bd\_print\_titles()

{

printf("%-8s", "Record #");

printf(" | ");

printf("%-8s", "ID");

printf(" | ");

printf("%-15s", "Train type");

printf(" | ");

printf("%-15s", "Departure Time");

printf(" | ");

printf("%-15s", "Wagon Quantity");

printf(" | ");

printf("%-12s", "Wagon number");

printf(" | ");

printf("%-15s", "Wagon type");

printf(" | ");

printf("%-12s", "Tickets");

printf(" | ");

printf("%-12s", "Wagon number");

printf(" | ");

printf("%-15s", "Wagon type");

printf(" | ");

printf("%-12s", "Tickets");

printf(" | ");

printf("\n");

for (int i = 0; i < 165; i++)

{

printf("-");

}

printf("\n");

}

void bd\_all\_write(char \*name)

{

FILE\* fout = fopen(name, "rb");

Tdata tr;

if (fout == NULL)

{

puts("Error while opening file.");

}

else

{

bd\_print\_titles();

while (!feof(fout))

{

fread(&tr, sizeof(Tdata), 1, fout);

if (!feof(fout))

{

write\_single\_item(tr);

}

}

puts("");

}

fclose(fout);

}

void bd\_add\_item(char\* name)

{

FILE\* fin = fopen(name, "r+b");

Tdata tr, temp;

int n, pos, i;

if (fin == NULL)

{

puts("Error while opening file");

}

else

{

fseek(fin, 0, SEEK\_END);

n = (int)ftell(fin) / sizeof(Tdata);

printf("Total database positions: %d\n", n);

printf("Enter push position: ");

scanf\_s("%d%\*c", &pos);

if (pos > n + 1)

{

puts("Error - unavailable position.");

}

else if (pos == n + 1)

{

tr = new\_item(pos);

fwrite(&tr, sizeof(Tdata), 1, fin);

}

else

{

tr = new\_item(pos);

fseek(fin, (n - 1) \* sizeof(Tdata), SEEK\_SET);

for (i = n; i >= pos; i--)

{

fread(&temp, sizeof(Tdata), 1, fin);

fseek(fin, i \* sizeof(Tdata), SEEK\_SET);

temp.number++;

fwrite(&temp, sizeof(Tdata), 1, fin);

fseek(fin, (i - 2) \* sizeof(Tdata), SEEK\_SET);

}

fseek(fin, i \* sizeof(Tdata), SEEK\_SET);

fwrite(&tr, sizeof(Tdata), 1, fin);

}

}

fclose(fin);

}

void bd\_write\_item\_and\_titles(char\* name)

{

FILE\* fout = fopen(name, "rb");

int n = 0, pos = 0;

Tdata tr;

if (fout == NULL)

{

puts("Error while opening file.\n");

}

else

{

fseek(fout, 0, SEEK\_END);

n = (int)ftell(fout) / sizeof(Tdata);

printf("Total database positions - %d\n", n);

printf("Enter number of record: ");

scanf\_s("%d%\*c", &pos);

if (pos > n)

{

puts("Error - unavailable number.\n");

}

else

{

puts("");

fseek(fout, (pos - 1) \* sizeof(Tdata), SEEK\_SET);

fread(&tr, sizeof(Tdata), 1, fout);

bd\_print\_titles();

write\_single\_item(tr);

}

}

fclose(fout);

}

**trains.h**

#pragma once

#include <stdio.h>

#define \_CRT\_SECURE\_NO\_WARNINGS

struct wagon{

unsigned int w\_number;

unsigned int wagon\_type;

unsigned int tickets;

};

typedef struct firstt{

unsigned int number;

unsigned int id;

unsigned int train\_type;

char time[10];

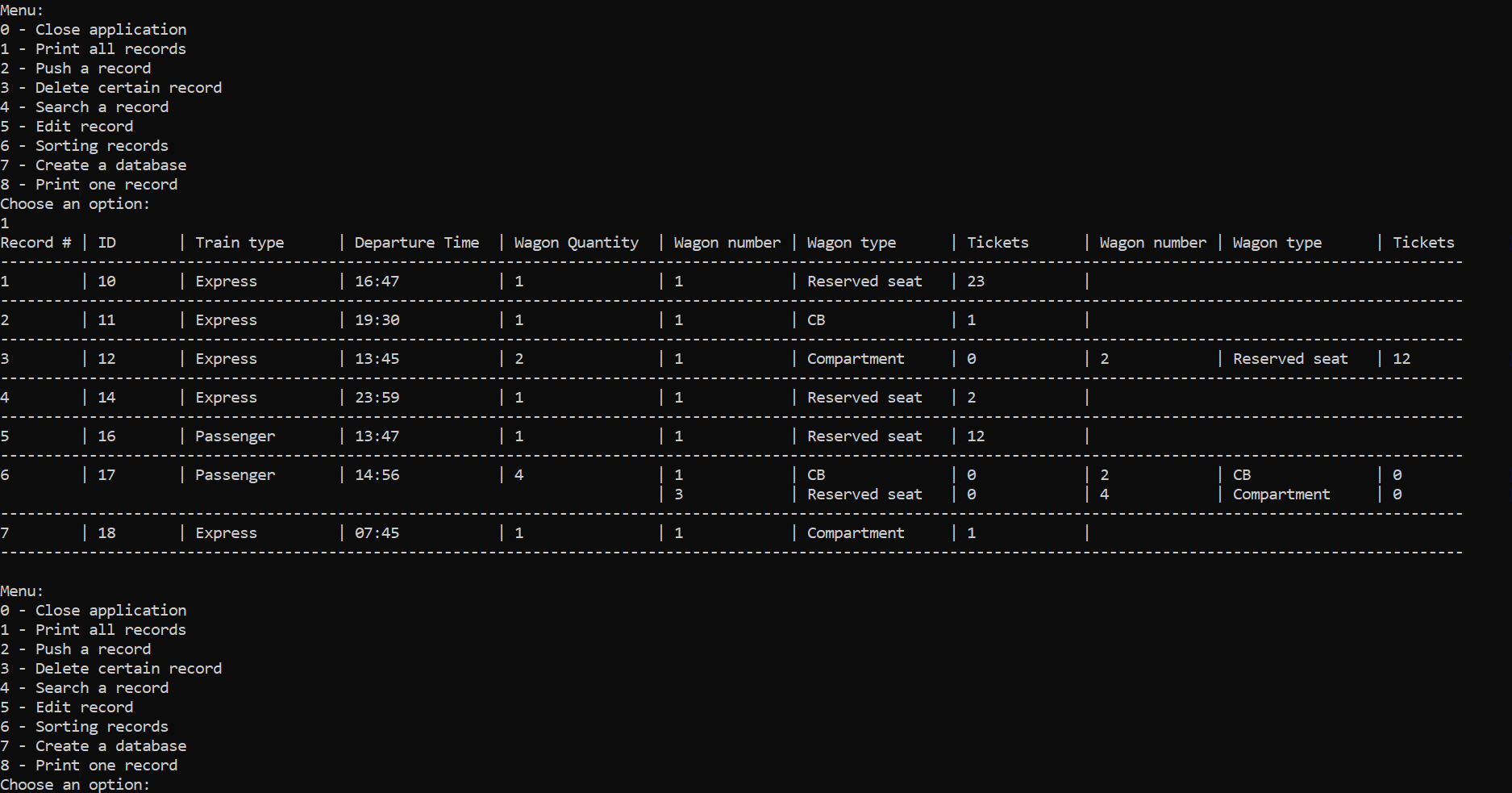
unsigned int wagon\_q;

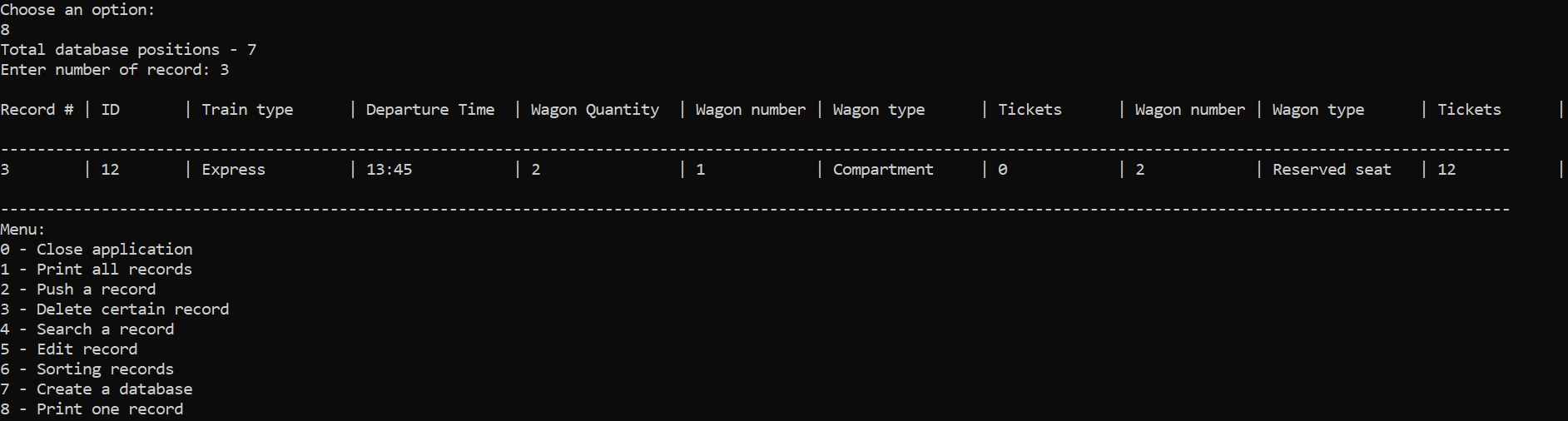
struct wagon wagons[15];

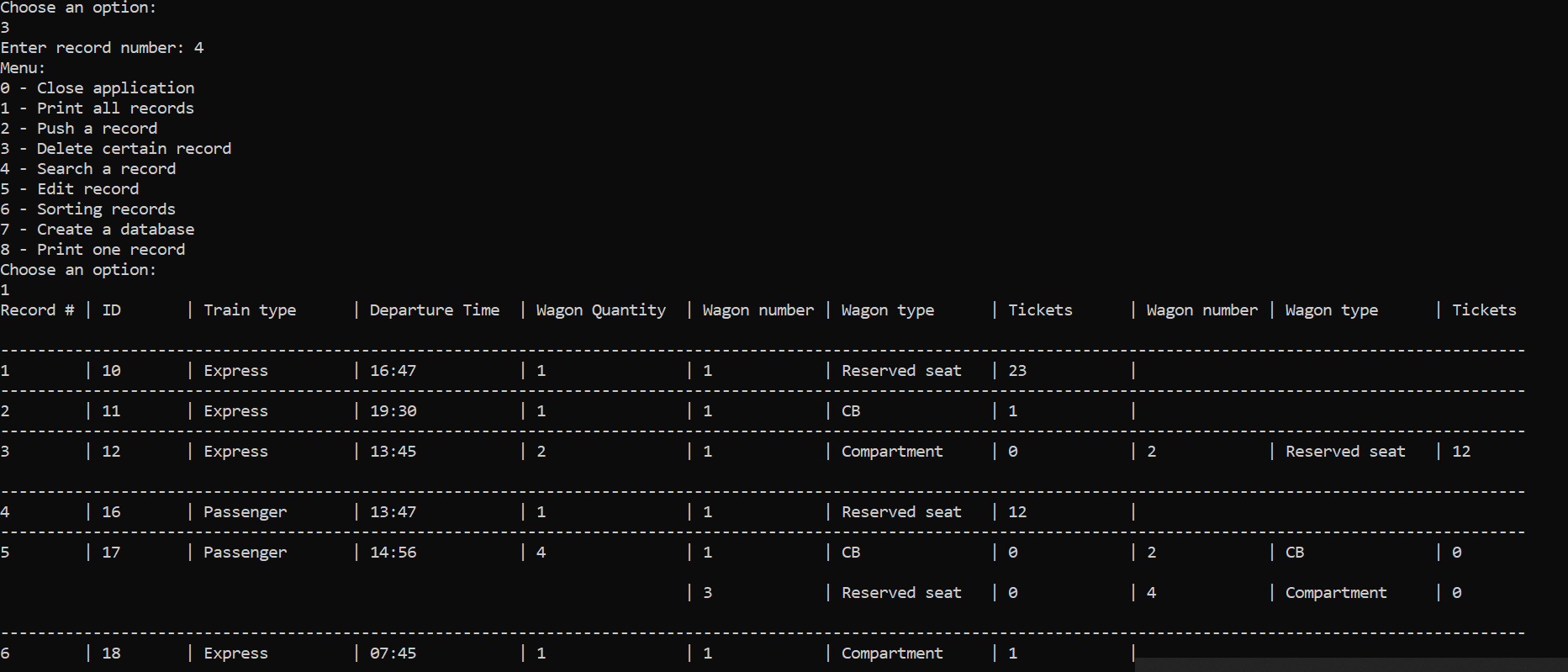
}train;

**ТЕСТЫ**

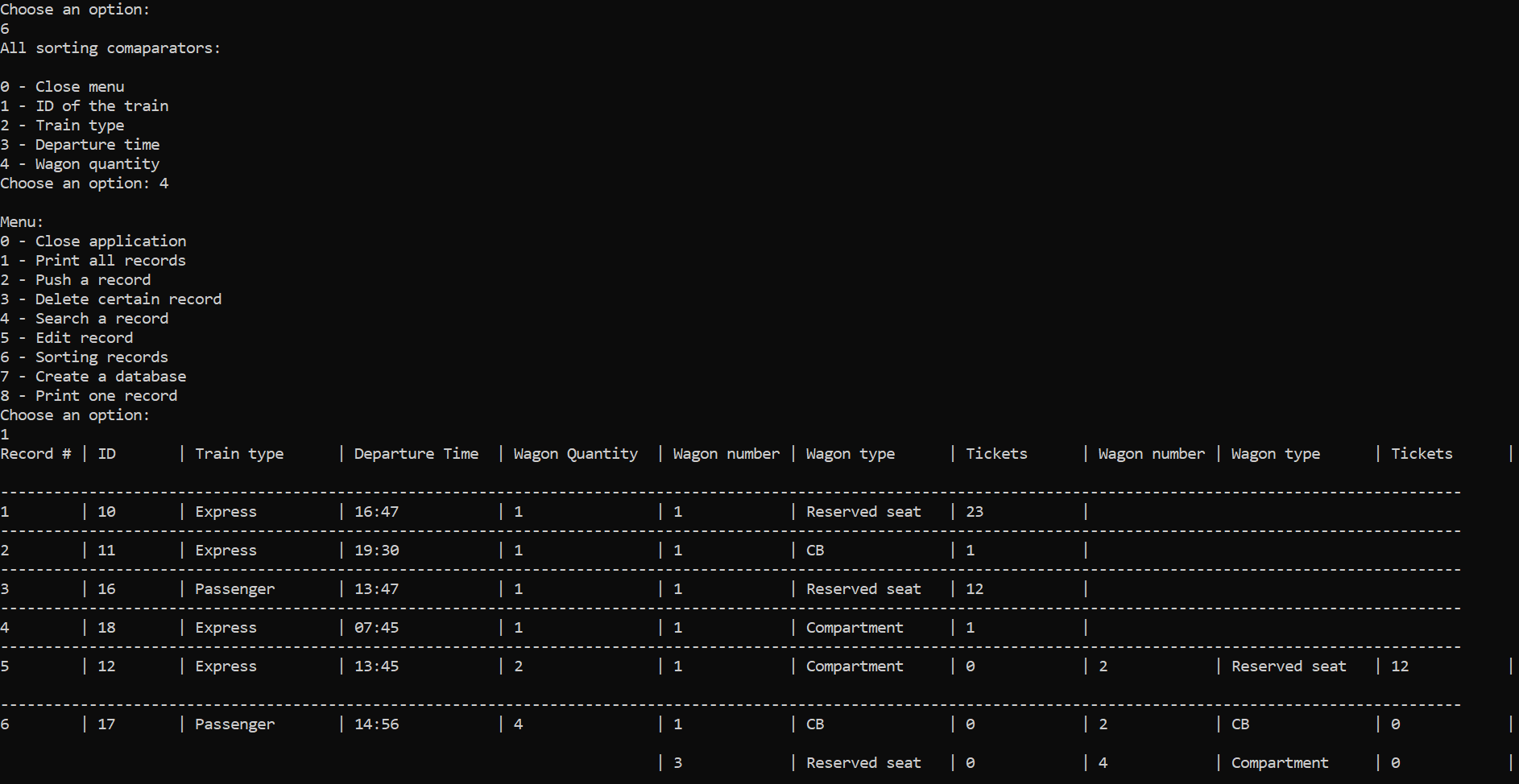
**Тест № 1**

****

**Тест № 2**

****

**Тест № 3**

****