Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Уфимский государственный нефтяной технический университет»

Факультет автоматизации производственных процессов

Кафедра вычислительной техники и инженерной кибернетики

ОТЧЕТ

по лабораторной работе №3

«Абстрактные типы данных»

Выполнил ст. гр. БПО-15-02 Гуватова Ф.Г.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

дата, подпись

Проверил: преподаватель Жолобова Г.Н.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

дата, подпись

*Целью* данной лабораторной работы является: изучение абстрактных типов данных, освоение навыков работы с указателями.

*Задание:*

Составить программу, которая содержит динамическую информацию о наличии автобусов в автобусном парке. Сведения о каждом автобусе включают:

* номер автобуса;
* фамилию и инициалы водителя
* номер маршрута;

Программа должна обеспечивать:

* начальное формирование данных обо всех автобусах в парке в виде списка;
* при выезде каждого автобуса из парка вводится номер автобуса, и программа удаляет данные об этом автобусе из списка атобусов, находящихся в парке, и записывает данные в список автобусов, находящиеся на маршруте;
* при въезде каждого автобуса в парк вводится номер автобуса, и программа удаляет данные об этом автобусе из списка автобусов, находящихся на маршруте, и записывает эти данные в список автобусов, находящихся в парке;
* по запросу выдаются сведения об автобусах, находящихся в парке, или об автобусах, находящихся на маршруте.

Для реализации поставленной задачи создадим пользовательскую программу в среде разработки Visual Studio.

Для реализации конкретно наших задач, создадим пользовательские функции add\_card (), которая заполняет список; send\_ret (), которая “отправляет” автобусы в маршрут или же в парк; show\_bus\_card(), которая указывает местоположение автобусов в данный момент (в парке или на маршруте).

Ниже на рисунках 1-5 предоставлена работа программы.

Код программы предоставлен в ***Приложении 1.***

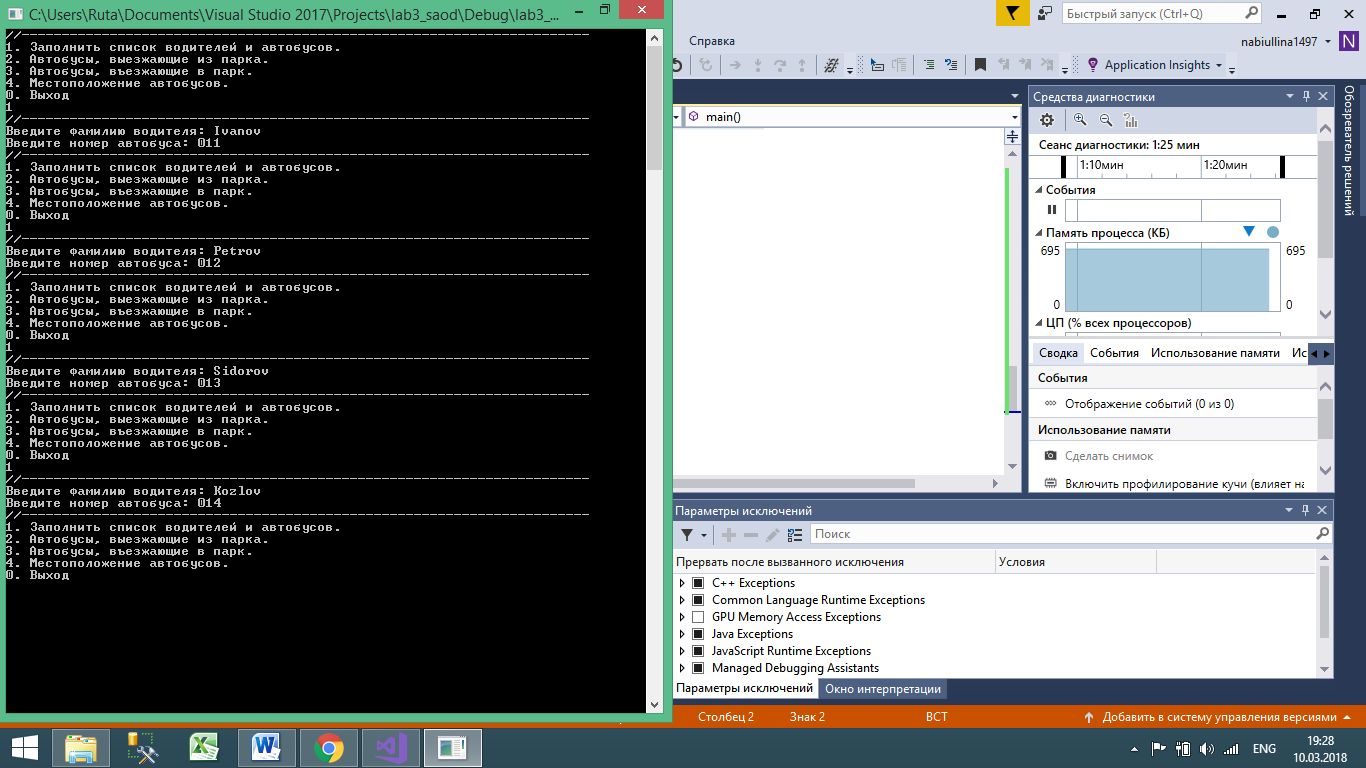


Рисунок 1 – Заполнение списка водителей и их автобусов.

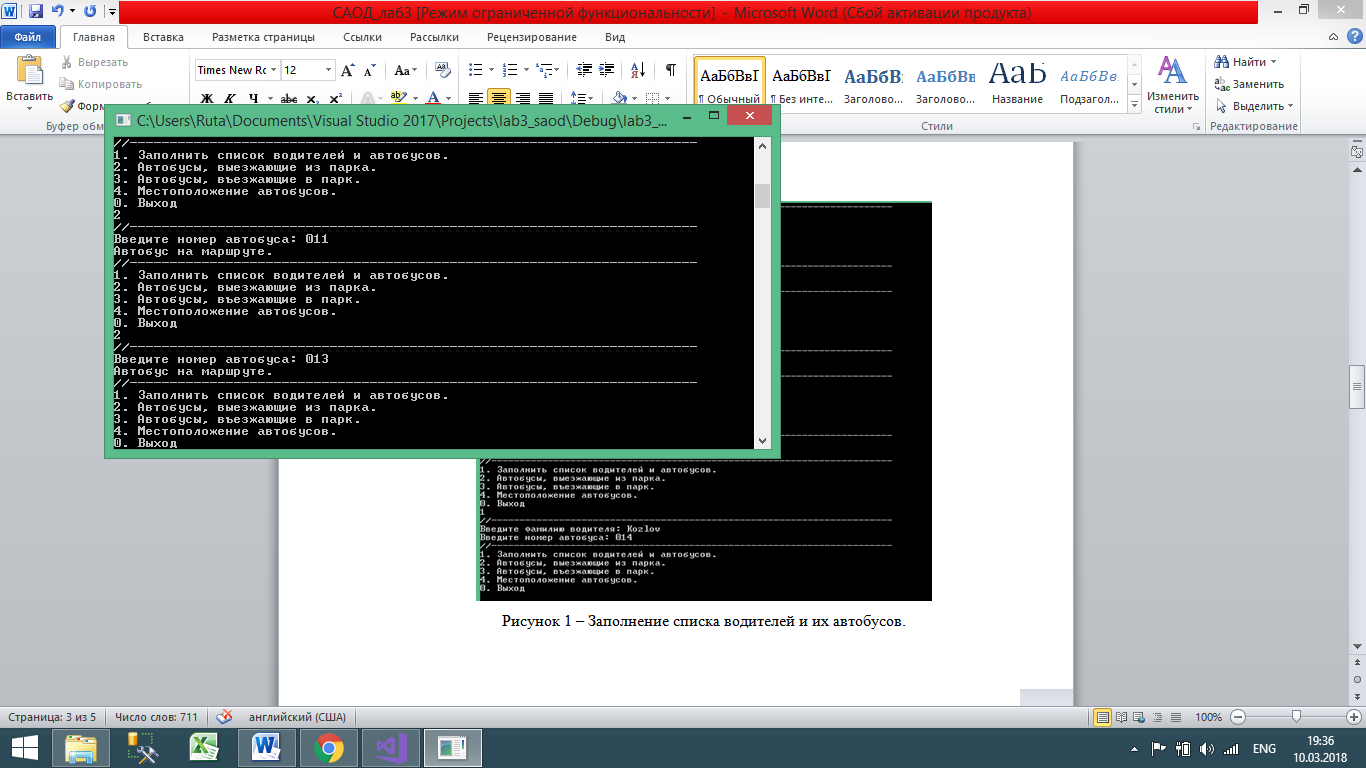


Рисунок – Выезд автобусов из парка.

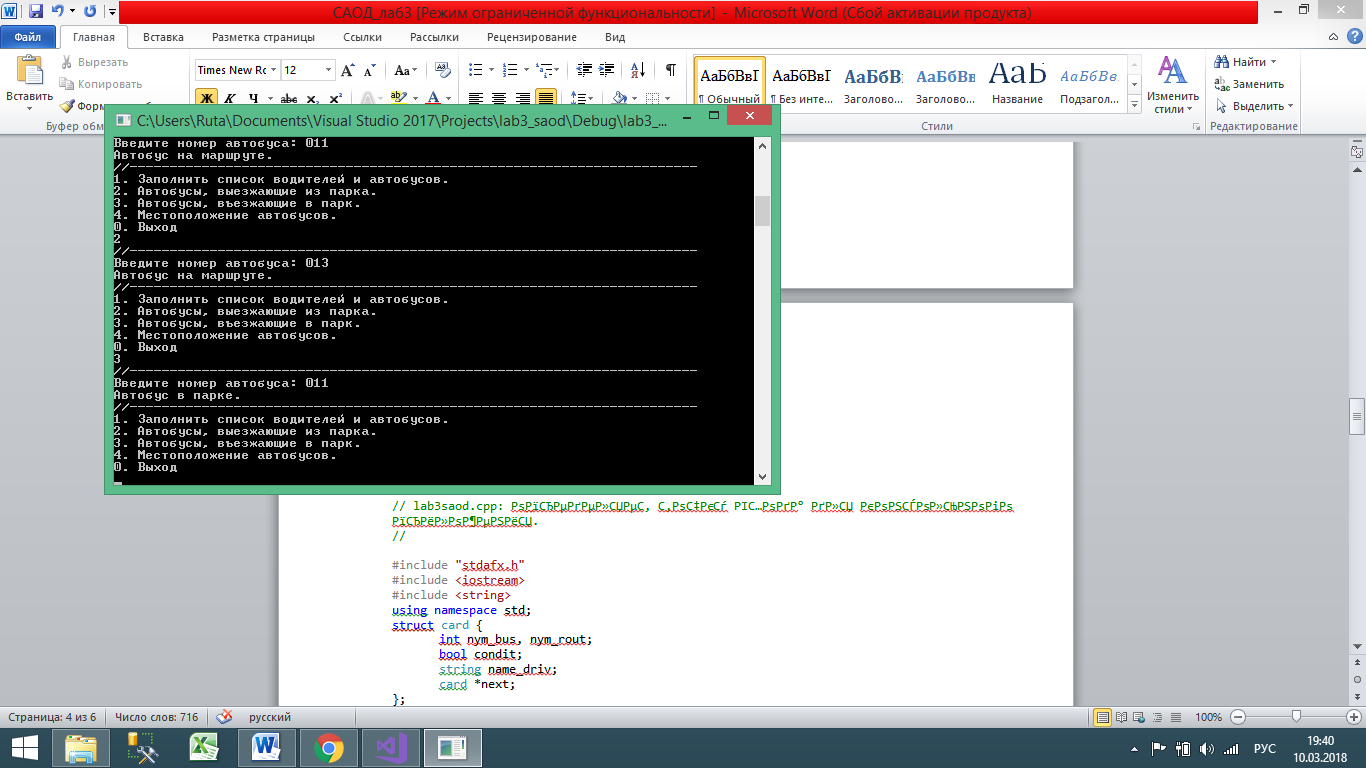


Рисунок 3 – Один автобус вернулся в парк.

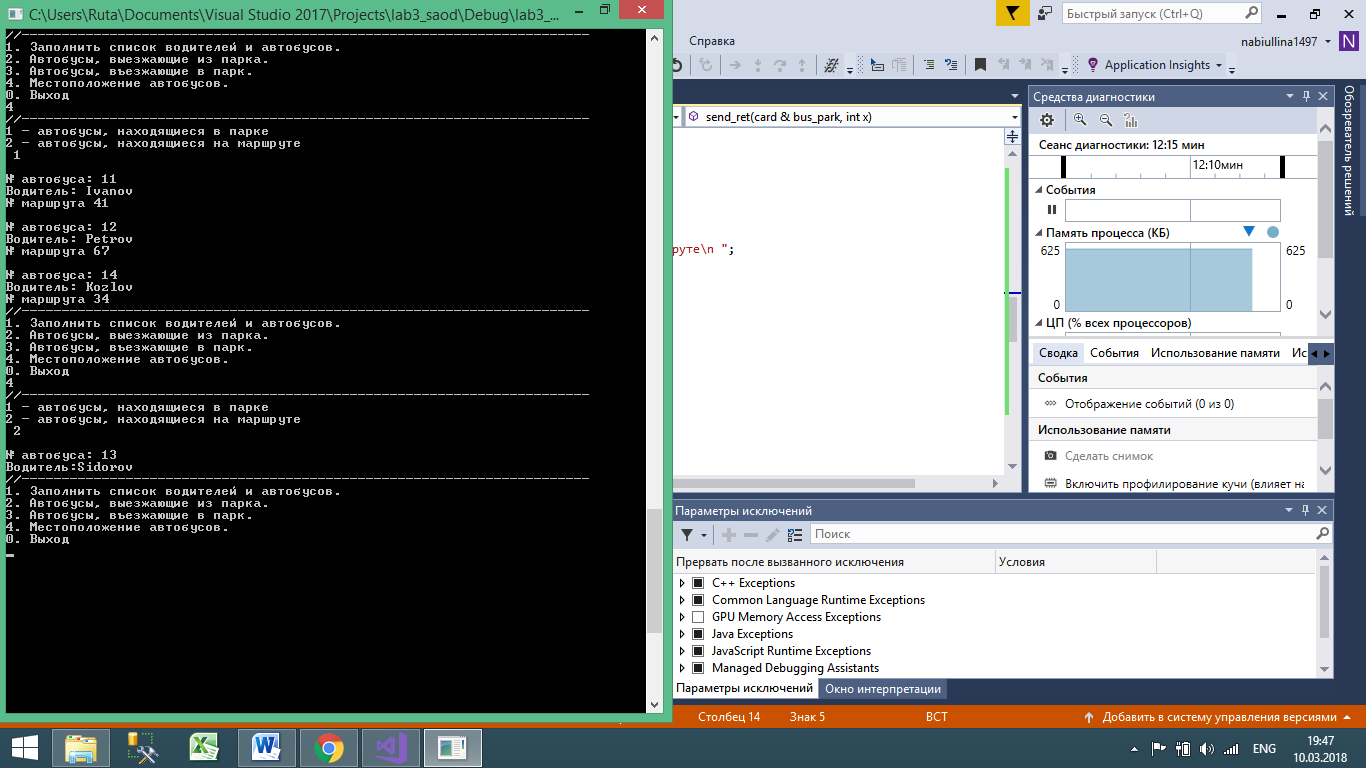


Рисунок 4 – Местоположение всех автобусов.

*Приложение 1.*

// lab3saod.cpp: РѕРїСЂРµРґРµР»СЏРµС‚ С‚РѕС‡РєСѓ РІС…РѕРґР° РґР»СЏ РєРѕРЅСЃРѕР»СЊРЅРѕРіРѕ РїСЂРёР»РѕР¶РµРЅРёСЏ.

//

#include "stdafx.h"

#include <iostream>

#include <string>

using namespace std;

struct card {

int nym\_bus, nym\_rout;

bool condit;

string name\_driv;

card \*next;

};

//-------------------------------------------------------------------------

void my\_cin(int &z) {

while (!(cin >> z) || (cin.peek() != '\n')) {

cin.clear();

while (cin.get() != '\n');

cout << "Ошибка ввода, попробуйте еще раз!" << endl;

}

}

//-------------------------------------------------------------------------

void add\_card(card \*&bus\_park) {

card \*ptr = new card;

cout << "Введите фамилию водителя: ";

cin >> ptr->name\_driv;

cout << "Введите номер автобуса: ";

my\_cin(ptr->nym\_bus);

ptr->nym\_rout = ptr->condit = 0;

ptr->next = nullptr;

if (bus\_park) {

for (auto ptr2 = bus\_park;; ptr2 = ptr2->next)

if (!ptr2->next) {

ptr2->next = ptr;

break;

}

}

else

bus\_park = ptr;

}

//-------------------------------------------------------------------------

void send\_ret(card &bus\_park, int x) {

int y;

cout << "Введите номер автобуса: "; my\_cin(y);

auto \*time\_ptr = &bus\_park;

for (; time\_ptr; time\_ptr = time\_ptr->next)

if (time\_ptr->nym\_bus == y)

break;

else if (!time\_ptr->next) {

cout << "Автобус под № " << y << " не обнаружен\n";

return;

}

/\*if (x == 2 && time\_ptr->condit || x == 3 && !time\_ptr->condit) {

if (x == 2)

cout << "(error) The bus is on the route.\n";

else

cout << "(error) The bus is in park.\n";

return;

}

else\*/

if (x == 2) {

time\_ptr->condit = 0;

cout << "Автобус в парке.\n";

}

else if (x ==3) {

time\_ptr->condit = 1;

time\_ptr->nym\_rout = rand() % 100;

cout << "Автобус на маршруте.\n";

}

}

//-------------------------------------------------------------------------

void show\_bus\_card(card \*&bus\_card) {

int c;

cout << "1 - автобусы, находящиеся в парке \n2 - автобусы, находящиеся на маршруте\n ";

my\_cin(c);

for (auto time\_ptr(bus\_card); time\_ptr; time\_ptr = time\_ptr->next)

if (c == 1 && time\_ptr->condit)

cout << "\n№ автобуса: " << time\_ptr->nym\_bus <<

"\nВодитель: " << time\_ptr->name\_driv <<

"\n№ маршрута " << time\_ptr->nym\_rout << endl;

else if (c == 2 && !time\_ptr->condit)

cout << "\n№ автобуса: " << time\_ptr->nym\_bus <<

"\nВодитель:" << time\_ptr->name\_driv << endl;

}

//-------------------------------------------------------------------------

int main()

{

setlocale(LC\_ALL, "Rus");

card \*bus\_park(nullptr);

int x(1);

while (x) {

cout << "//-----------------------------------------------------------------------\n";

cout << "1. Заполнить список водителей и автобусов.\n";

cout << "2. Автобусы, выезжающие из парка.\n";

cout << "3. Автобусы, въезжающие в парк.\n";

cout << "4. Местоположение автобусов.\n";

cout << "0. Выход\n";

my\_cin(x);

cout << "//-----------------------------------------------------------------------\n";

switch (x) {

case 1:

add\_card(bus\_park);

break;

case 2:

if (bus\_park) send\_ret(\*bus\_park, x);

else cout << "Чтобы начать, заполните список!\n";

break;

case 3:

if (bus\_park) send\_ret(\*bus\_park, x);

else cout << "Чтобы начать, заполните список!\n";

break;

case 4:

if (bus\_park) show\_bus\_card(bus\_park);

else cout << "Чтобы начать, заполните список!\n";

break;

default:

if (x)

cout << "Введите номер от 0 до 4\n";

}

}

return 0;

}

**Вывод:** приложение, разработанное в ходе выполнения лабораторной работы, функционирует корректно и справляется со всеми поставленными задачами. Значит, задание выполнено верно.