**Министерство образования и науки Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**

**МЕХАНИКИ И ОПТИКИ”**

**ФАКУЛЬТЕТ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

**ОТЧЕТ**

**по Лабораторной работе № 7**

**«РАБОТА С ДИНАМИЧЕСКИМИ СТРУКТУРАМИ ДАННЫХ»**

Специальность 09.02.03«Программирование в компьютерных системах»

Дисциплина «Прикладное программирование»

|  |  |
| --- | --- |
| Преподаватель:  Сорокин Д.Ю.  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2018г.  Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Выполнил:  студент группы Y2335  Котлицкий С.А. |

Санкт-Петербург

2018/2019

**Цель работы:** продемонстрировать работа с динамическими структурами данных, на примере списков, очередей, стеков или бинарных деревьев

**Практическое задание:** реализовать программу согласно варианту задания

**Индивидуальное задание:**

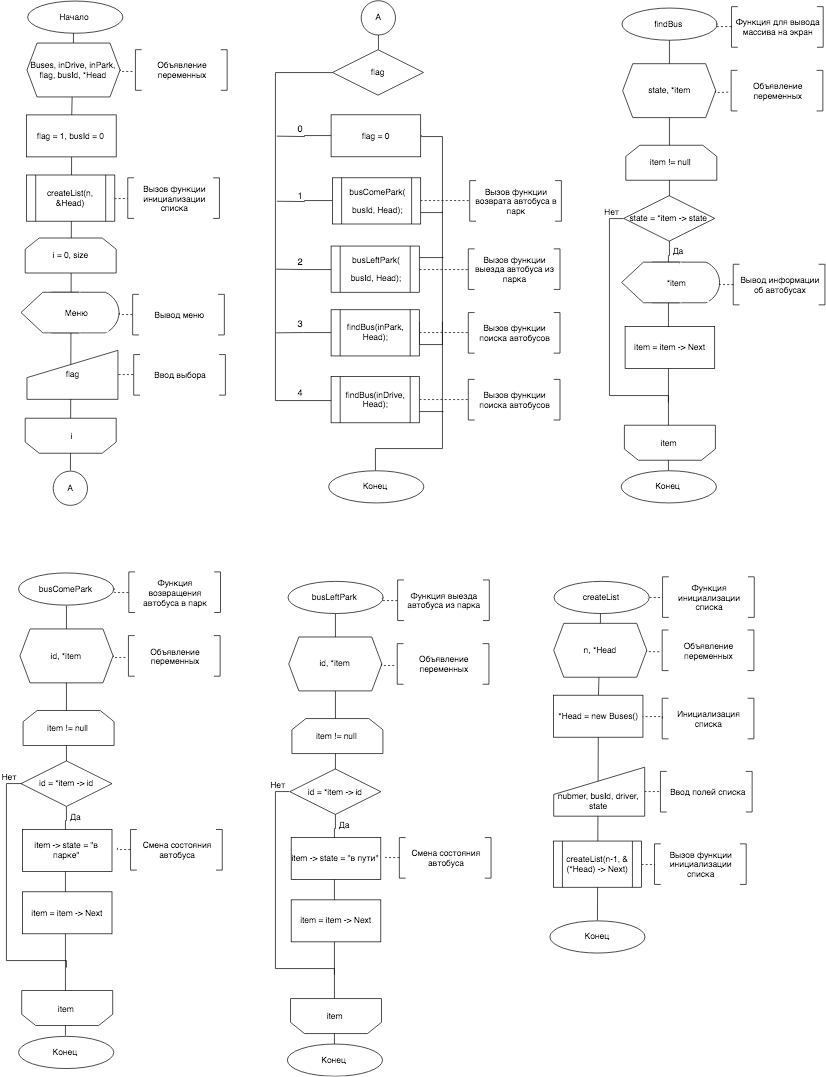
Составить программу, которая содержит динамическую информацию о наличии автобусов в автобусном парке. Сведения о каждом автобусе включают:

1. номер автобуса;
2. фамилию и инициалы водителя;
3. номер маршрута;
4. признак того, где находится автобус — на маршруте или в парке.

Программа должна обеспечивать:

1. начальное формирование данных обо всех автобусах в виде списка;
2. при выезде каждого автобуса из парка вводится номер автобуса, и программа устанавливает значение признака «автобус на маршруте»;
3. при въезде каждого автобуса в парк вводится номер автобуса, и программа устанавливает значение признака «автобус в парке»;
4. по запросу выдаются сведения об автобусах, находящихся в парке, или об автобусах, находящихся на маршруте.

Блок-схема алгоритма работы программы представлена ниже (Рисунок 1):



*Рисунок 1 Блок-схема алгоритма программы*

Код программы:

#include <iostream>//Подключаем библиотеки

#include <istream>

#include <string>

using namespace std;

struct Buses {//Объявляем список

int busId, number;

string driver;

string state; //в парке | в пути

Buses \*Next;

};

Buses \*Head;//Указатель на начало

void createList (int n, Buses\*\* Head) {//Функция создания списка

if (n > 0) {

(\*Head) = new Buses();

//Ввод значений

cout << "Введите номер маршрута" << endl;

cin >> (\*Head) -> number;

cout << "Введите Фамилия И.О. водителя" << endl;

cin.ignore();

getline(cin, (\*Head) -> driver);

cout << "Введите номер автобуса" << endl;

cin >> (\*Head) -> busId;

cout << "Введите статус автобуса (в пути / в парке)" << endl;

cin.ignore();

getline(cin, (\*Head) -> state);

//Обнуление указателя на следующий элемент

(\*Head) -> Next = NULL;

createList(n-1, &((\*Head) -> Next));

}

}

void busComePark (int id, Buses\* Head) {//Функция возврата автобуса в парк

Buses \*item;

item = Head;

while (item != NULL) {

if (id == item -> busId) {//Проверка по условию

item -> state = "в парке";

}

item = item -> Next;

}

}

void busLeftPark (int id, Buses\* Head) {//Функция выезда автобуса из парка

Buses \*item;

item = Head;

while (item != NULL) {

if (id == item -> busId) {//Проверка по условию

item -> state = "в пути";

}

item = item -> Next;

}

}

void findBus (string state, Buses\* Head) {//Функция поиска автобуса

Buses \*item;

item = Head;

cout << "Список автобусов " << state << ':' << endl;

while (item != NULL) {

if (state.compare((item -> state)) == 0) {//Проверка по условию

//Вывод значений

cout << endl << "\t номер автобуса - " << item -> busId << endl;

cout << "\t номер маршрута - " << item -> number << endl;

cout << "\t ФИО водителя - " << item -> driver << endl;

}

item = item -> Next;

}

}

int main(int argc, const char \* argv[]) {

setlocale(LC\_ALL, "Russian");

//Инициализация списка

createList(2, &Head);

string inDrive = "в пути", inPark = "в парке";

int flag = 1, busId = 0;

while (flag != 0) {

//Вывод меню

cout << endl << "Выбор действия:\n"

<< "Автобус въехал в парк\t--> 1\n"

<< "Автобус выехал из парка\t--> 2\n"

<< "Сведения об автобусах:\n"

<< " Автобусы в парке\t\t--> 3\n"

<< " Автобусы в пути\t\t--> 4\n"

<< "Выход из программы\t --> 0" << endl ;

cin >> flag;

switch (flag){ //Выбор действия

case 0:

flag = 0;

break;

case 1:

cout << "Введите номер автобуса" << endl;

cin >> busId;

busComePark(busId, Head);

break;

case 2:

cout << "Введите номер автобуса" << endl;

cin >> busId;

busLeftPark(busId, Head);

break;

case 3:

findBus(inPark, Head);

break;

case 4:

findBus(inDrive, Head);

break;

}

}

return 0;

}

Протокол программы:

Введите номер маршрута

87

Введите Фамилия И.О. водителя

Кириллов В.С.

Введите номер автобуса

4321

Введите статус автобуса (в пути / в парке)

в пути

Введите номер маршрута

45

Введите Фамилия И.О. водителя

Сидоров А.С.

Введите номер автобуса

8732

Введите статус автобуса (в пути / в парке)

в парке

Выбор действия:

Автобус въехал в парк --> 1

Автобус выехал из парка --> 2

Сведения об автобусах:

Автобусы в парке --> 3

Автобусы в пути --> 4

Выход из программы --> 0

3

Список автобусов в парке:

номер автобуса - 8732

номер маршрута - 45

ФИО водителя - Сидоров А.С.

Выбор действия:

Автобус въехал в парк --> 1

Автобус выехал из парка --> 2

Сведения об автобусах:

Автобусы в парке --> 3

Автобусы в пути --> 4

Выход из программы --> 0

4

Список автобусов в пути:

номер автобуса - 4321

номер маршрута - 87

ФИО водителя - Кириллов В.С.

Выбор действия:

Автобус въехал в парк --> 1

Автобус выехал из парка --> 2

Сведения об автобусах:

Автобусы в парке --> 3

Автобусы в пути --> 4

Выход из программы --> 0

2

Введите номер автобуса

8732

Выбор действия:

Автобус въехал в парк --> 1

Автобус выехал из парка --> 2

Сведения об автобусах:

Автобусы в парке --> 3

Автобусы в пути --> 4

Выход из программы --> 0

4

Список автобусов в пути:

номер автобуса - 4321

номер маршрута - 87

ФИО водителя - Кириллов В.С.

номер автобуса - 8732

номер маршрута - 45

ФИО водителя - Сидоров А.С.

Выбор действия:

Автобус въехал в парк --> 1

Автобус выехал из парка --> 2

Сведения об автобусах:

Автобусы в парке --> 3

Автобусы в пути --> 4

Выход из программы --> 0

0

Program ended with exit code: 0