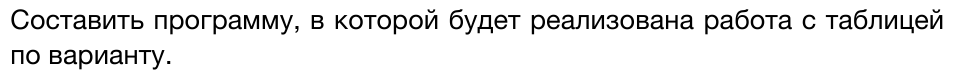
**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ 6**

**Общее задание**

****

**Пример решения**

#include <stdlib.h>

#include <cstdlib>

#include <iostream>

#include <clocale>

#include <cmath>

#include <ctime>

#include <iomanip>

#include <conio.h>

using namespace std;

int Size = 3; //глобальная переменная, доступна в любом месте программы - размер массива

struct bus { // создаем структуру

string view\_transport;

string view\_route;

float long\_S;

float time;

};

void printRow( bus array) //формирование границ и вывод

{

cout << " | " << array.view\_transport << setw(17 - array.view\_transport.length()) //создаем красивый вывод таблицы, формируем нужное количество отступов от названий

<< " | " << array.view\_route << setw(13 - array.view\_route.length()) //если имя содержит один символ, мы отступаем 14 символов, если два - 12 и т.д.

<< " | " << setw(26) << array.long\_S

<< " | " << setw(19) << array.time

<< " | " << endl;

cout << " ----------------------------------------------------------------------------------" << endl;

}

void printHeader() //печатаем заголовок

{

cout << " ----------------------------------------------------------------------------------" << endl;

cout << " | ВЕДОМОСТЬ ОБЩЕСТВЕННОГО ТРАНСПОРТА |" <<

endl;

cout << " ----------------------------------------------------------------------------------" << endl;

cout << " | Вид транспорта | № маршрута | Протяженность маршрута (км) | Время в дороге (мин)) | " << endl;

cout << " ----------------------------------------------------------------------------------" << endl;

}

void printTable(bus\* array) //Вывести таблицу

{

system("CLS"); //очистка терминала

printHeader(); //печатаем заголовок

for (int i = 0; i < Size; i++) //выводим на экран

{

printRow(array[i]);

}

}

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

setlocale(LC\_ALL, "Russian");

bus\* array = new bus[Size];//создание динамического массива с данными структур

array[0] = { "Тр", " 2", 29.5, 90 };

array[1] = { "Тс", "71K", 16.2, 50};

array[2] = { "Ас", " 35", 60, 129 };

int action=999;

printHeader();

printTable(array);

while (action != 0) {

cout << "Выберите действие над таблицей" << '\n';

cout << "Введите 1, если хотите добавить строку " << '\n';

cout << "Введите 2, если хотите изменить строку " << '\n';

cout << "Введите 3, если хотите удалить строку " << '\n';

cout << "Введите 0, если хотите завершить работу с таблицей " << '\n';

cin >> action;

if (action == 1) {

printHeader();

printTable(array);

Size++;

bus\* newarray = new bus[Size];

for (int i = 0; i < Size-1; i++)

newarray[i] = array[i];

delete[] array; //Удаляем старый массив

array = newarray;

cout << "Введите название транспорта, который хотите добавить- ";

cin >> newarray[Size-1].view\_transport;

cout << "Введите номер маршрута- ";

cin >> newarray[Size-1].view\_route;

cout << "Введите длину маршрута- ";

cin >> newarray[Size-1].long\_S;

cout << "Введите время в пути- ";

cin >> newarray[Size-1].time;

system("CLS"); //очистка терминала

printHeader();

bus\* array = new bus[Size];//создание динамического массива с данными структур

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

array[i] = newarray[i];

printTable(array);

}

//..

if (action == 2) {

bus\* newarray = new bus[Size];

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

newarray[i] = array[i];

delete[] array; //удаляем старый массив

cout << "Введите номер строки, которую хотите изменить- " << '\n';

int n;

cin >> n;

cout << "Введите название транспорта, который хотите добавить- ";

cin >> newarray[n - 1].view\_transport;

cout << "Введите номер маршрута- ";

cin >> newarray[n - 1].view\_route;

cout << "Введите длину маршрута- ";

cin >> newarray[n - 1].long\_S;

cout << "Введите время в пути- ";

cin >> newarray[n - 1].time;

system("CLS"); //очистка терминала

array = newarray;

printHeader();

printTable(array);

}

if (action == 3) {

bus\* newarray = new bus[Size];

int DeleteIndex = 0;

cout << "Введите номер строки, которую хотите удалить- ";

cin >> DeleteIndex;

DeleteIndex--; //преобразуем номер строки в индекс

int k = 0;

for (int i = 0; i < Size; i++) //пробегаем по всем объектам старого массива

{

if (DeleteIndex != i) {

newarray[k] = array[i];

k++;

}

}

Size--;

array = new bus[Size];

array = newarray;

printHeader();

printTable(array);

}

}

}

**Результат работы**

Выберем действие с таблицей.

