/\*III - 3. Разработать контейнерный класс «Депо», в котором список вагонов упорядочен по возрастанию

значений поля Номер. Проверить, есть ли в депо пассажирские вагоны типа купе с количеством

мест менее 36? Если есть, то распечатать информацию о вагоне с наименьшим номером.

Использовать STL контейнер list\*/

#include <iostream>

#include <fstream>

#include <Windows.h>

#include "Depot.h"

using std::cout;

using std::ifstream;

bool Coupe\_Carriage\_with\_seats\_less\_than\_36\_\_is\_exist(Depot& Depo);

int main()

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

bool Error = false;

ifstream input("input.txt");

if (!input)

{

cout << "Файл input.txt не был открыт!\n";

system("pause");

Error = true;

}

if (!Error)

{

Depot Depo(input);

int choice;

bool ext = false;

while (!ext)

{

cout << "Введите номер необходимого пункта\n";

cout << "1. Выполнение программы.\n";

cout << "0. Завершение программы.\n\n";

choice = \_getwch();

switch (choice)

{

case '1': //

{

Depo.print();

cout << '\n';

if (!Coupe\_Carriage\_with\_seats\_less\_than\_36\_\_is\_exist(Depo))

cout << "Нужного вагона в " << Depo.get\_name() << " не нашлось.";

cout << '\n';

cout << '\n';

}

break; //

case '0': //Завершение программы

ext = 1;

break; //Завершение программы

default:

cout << "Неверный формат ввода, попробуйте ещё раз\n\n";

break;

}

}

}

input.close();

return 0;

}

//=================================================================================

bool Coupe\_Carriage\_with\_seats\_less\_than\_36\_\_is\_exist(Depot& Depo)

{

bool result = false;

list<Carriage\*>::iterator iter = Depo.begin();

Passenger\_Carriage\* tmp = nullptr;

while (iter != Depo.end() && !result)

{

tmp = dynamic\_cast<Passenger\_Carriage\*>(\*iter);

if (tmp && (\*tmp).get\_type() == Passenger\_Carriage::Coupe && (\*tmp).get\_number\_of\_seats() < 36)

{

result = true;

cout << "Нужный вагон:";

cout << "\n==================================\n";

(\*\*iter).print();

}

++iter;

}

tmp = nullptr;

return result;

}