**Курс «Объектно-ориентированное программирование на C++»**

**Встреча №30**

Тема: STL

Задание для самостоятельной работы.

**Задание.**

Написать программу «Автоматизированная информационная система ЖД вокзала». Система содержит: сведения об отправлении поездов дальнего следования. Для каждого поезда указываем: номер, время отправления, станцию назначения. Обеспечить:

■■ Ввод данных в систему;

■■ Вывод информации о всех поездах;

■■ Вывод информации о запрашиваемом поезде.

**Использовать контейнерный класс vector**

**#define** \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include<string>

#include<time.h>

#include<vector>

#include<Windows.h>

using namespace std;

class Train

{

private:

string number;

string departure\_time;

string end\_place;

public:

string& Get\_Number()

{

return number;

}

Train(string number, string departure\_time, string end\_place)

{

this->number = number;

this->departure\_time = departure\_time;

this->end\_place = end\_place;

}

friend ostream& operator<<(ostream& os, const Train& t);

};

ostream& operator<<(ostream& os, const Train& t)

{

os << "Номер потяга: "<<t.number <<endl;

os << "Час відправлення: " << t.departure\_time << endl;

os << "Потяг прямує на станцію " << t.end\_place << endl;

os << "=========================================\n";

return os;

}

class TrainStation

{

private:

vector<Train> arr;

public:

void Add\_Train(string number, string departure\_time, string end\_place)

{

arr.push\_back(Train(number, departure\_time, end\_place));

}

void Show\_All()

{

if (arr.size())

{

for (auto it = arr.begin(); it != arr.end(); it++)

cout << \*it;

}

else

cout << "Немає потягів!\n";

}

Train Search\_By\_Nomer(string number)

{

for (int i = 0; i < arr.size(); i++)

{

if (arr[i].Get\_Number() == number)

return arr[i];

}

cout << "Немає такого елемента буде повернутий нульовий\n";

return arr[0];

}

int Get\_Size()

{

return arr.size();

}

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(0));

int k, count;

TrainStation station;

string number;

string departure\_time;

string end\_place;

cout << "Вокзал\n";

cout << "Додати потягів ->1, вивести весь список ->2, знайти потяг по номеру ->3, вихід ->0\n";

cin >> k;

while (k)

{

switch (k)

{

case 1:

cout << "Скільки потягів додати ->";

cin >> count;

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

{

cout << i + 1 << " потяг\n";

cout << "Введіть номер потяга->";

cin >> number;

cout << "Введіть час відправлення->";

cin >> departure\_time;

cout << "Введіть куди прямує(станція)->";

cin >> end\_place;

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

station.Add\_Train(number, departure\_time, end\_place);

}

break;

case 2:

station.Show\_All();

break;

case 3:

if (station.Get\_Size())

{

cout << "Введіть номер потяга, якого треба знайти->";

cin >> number;

cout << station.Search\_By\_Nomer(number);

}

else

cout << "Немає потягів!\n";

break;

}

cout << "Додати потягів ->1, вивести весь список ->2, знайти потяг по номеру ->3, вихід ->0\n";

cin >> k;

}

return 0;

}