**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

**КЫРГЫЗСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСТИТЕТ**

**им. И.Раззакова**

**ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

Кафедра «**Программное обеспечение компьютерных систем**»

Направление: 710400 «**Программная инженерия**»

Дисциплина: ««**Объектно-ориентированное программирование**»»

ОТЧЕТ

Лабораторная работа №4

Выполнил: Кудайбердиев Эрлан

Группа: ПИ-2-21

Проверил: Мусабаев Э. Б.

Бишкек – 2024

**Задание №1**

Для участия в ежегодном трансконтинентальном Ралли-марафоне «Дакар» по усложненной трассе в Южной Америке выбраны грузовики [КАМАЗ](https://ru.wikipedia.org/wiki/%D0%9A%D0%90%D0%9C%D0%90%D0%97) и [Tatra](https://ru.wikipedia.org/wiki/Tatra), которые после 2000 года лидируют в зачете грузовиков.

Решите задачу сравнения скоростей движения грузовиков по разным по проходимости участкам трассы, а именно: по равнине, горам, пустыне. Создайте и выдайте на экран таблицу результатов ралли - марафона. Определите победителя.

Для решения задачи используйте классы **Kamaz** и **Tatra,** а такжефункциюсравнения скоростей **FrCreater (**). Функция  **FrCreater (**) возвращает число +1, если объект **kamaz** движется быстрее объекта **tatra**; нуль, если их скорости одинаковы; число -1, если объект **kamaz** движется медленнее объекта **tatra** .

Оба класса содержат поля: «скорость» и «наименование» грузовой машины, а также методы: инициализация и отображение полей на экране. Определитесь с идентификаторами доступа к членам класса, не нарушая принципа инкапсуляции.

**Код программы**

#include <iostream>

#include <string>

using namespace std;

class Kamaz {

private:

string name;

double speed;

public:

Kamaz(const string& n, double s){

name = n;

speed = s;

}

void display(){

cout << "Скорость камаза '" << name << "' равна " << speed << " км/час" << endl;

}

double getSpeed() const{

return speed;

}

string getName() const {

return name;

}

};

class Tatra {

private:

string name;

double speed;

public:

Tatra(const string& n, double s){

name = n;

speed = s;

}

void display() const {

cout << "Скорость татры '" << name << "' равна " << speed << " км/час" << endl;

}

double getSpeed() const {

return speed;

}

string getName() const {

return name;

}

};

int FrCreator(const Kamaz& kamaz, const Tatra& tatra) {

if (kamaz.getSpeed() > tatra.getSpeed()) {

return 1;

}

else if (kamaz.getSpeed() < tatra.getSpeed()) {

return -1;

}

else {

return 0;

}

}

int main() {

setlocale(LC\_ALL, "rus");

Kamaz kamaz("камаз1", 80);

Tatra tatra("татра1", 90);

int result = FrCreator(kamaz, tatra);

if (result == 1) {

kamaz.display();

tatra.display();

cout << endl;

cout <<kamaz.getName() << " быстрее чем " <<tatra.getName() <<"!!!" << endl;

}

else if (result == -1) {

kamaz.display();

tatra.display();

cout << endl;

cout << tatra.getName() << " быстрее чем " << kamaz.getName() <<"!!!" << endl;

}

else {

kamaz.display();

tatra.display();

cout << endl;

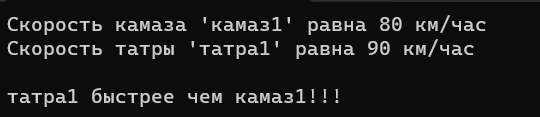
cout << "Их скорости одинаковые!!!" << endl;

}

return 0;

}

**Результат:**



**Задание №2**

Сделайте функцию **FrCreater(**) дружественной: классу **Kamaz,** обоим классам **Kamaz** и **Tatra.**

**Код программы**

#include <iostream>

#include <string>

using namespace std;

class Tatra;

class Kamaz {

private:

string name;

double speed;

public:

Kamaz(const string& n, double s) : name(n), speed(s) {}

void display() const {

cout << "Скорость камаза '" << name << "' равна " << speed << " км/час" << endl;

}

double getSpeed() const {

return speed;

}

string getName() const {

return name;

}

friend int FrCreator(const Kamaz& kamaz, const Tatra& tatra);

};

class Tatra {

private:

string name;

double speed;

public:

Tatra(const string& n, double s) : name(n), speed(s) {}

void display() const {

cout << "Скорость татры '" << name << "' равна " << speed << " км/час" << endl;

}

double getSpeed() const {

return speed;

}

string getName() const {

return name;

}

friend int FrCreator(const Kamaz& kamaz, const Tatra& tatra);

};

int FrCreator(const Kamaz& kamaz, const Tatra& tatra) {

if (kamaz.speed > tatra.speed) {

return 1;

}

else if (kamaz.speed < tatra.speed) {

return -1;

}

else {

return 0;

}

}

int main() {

setlocale(LC\_ALL, "rus");

Kamaz kamaz("камаз1", 80);

Tatra tatra("татра1", 90);

int result = FrCreator(kamaz, tatra);

if (result == 1) {

kamaz.display();

tatra.display();

cout << endl;

cout << kamaz.getName() << " быстрее чем " << tatra.getName() << "!!!" << endl;

}

else if (result == -1) {

kamaz.display();

tatra.display();

cout << endl;

cout << tatra.getName() << " быстрее чем " << kamaz.getName() << "!!!" << endl;

}

else {

kamaz.display();

tatra.display();

cout << endl;

cout << "Их скорости одинаковые!!!" << endl;

}

return 0;

}

Здесь функция FrCreator() объявлена как дружественная для классов Kamaz и Tatra, что позволяет ей иметь доступ к приватным членам обоих классов.

**Задание №3**

Сделайте класс **Tatra** дружественным классу **Kamaz.**

**Код программы**

#include <iostream>

#include <string>

using namespace std;

class Tatra;

class Kamaz {

private:

string name;

double speed;

public:

Kamaz(const string& n, double s) : name(n), speed(s) {}

void display() const {

cout << "Скорость камаза '" << name << "' равна " << speed << " км/час" << endl;

}

double getSpeed() const {

return speed;

}

string getName() const {

return name;

}

friend class Tatra;

friend int FrCreator(const Kamaz& kamaz, const Tatra& tatra);

};

class Tatra {

private:

string name;

double speed;

string j;

public:

Tatra(const string& n, double s) : name(n), speed(s) {}

void display() const {

cout << "Скорость татры '" << name << "' равна " << speed << " км/час" << endl;

}

double getSpeed() const {

return speed;

}

string getName() const {

return name;

}

friend int FrCreator(const Kamaz& kamaz, const Tatra& tatra);

};

int FrCreator(const Kamaz& kamaz, const Tatra& tatra) {

if (kamaz.speed > tatra.speed) {

return 1;

}

else if (kamaz.speed < tatra.speed) {

return -1;

}

else {

return 0;

}

}

int main() {

setlocale(LC\_ALL, "rus");

Kamaz kamaz("камаз1", 80);

Tatra tatra("татра1", 90);

int result = FrCreator(kamaz, tatra);

if (result == 1) {

kamaz.display();

tatra.display();

cout << endl;

cout << kamaz.getName() << " быстрее чем " << tatra.getName() << "!!!" << endl;

}

else if (result == -1) {

kamaz.display();

tatra.display();

cout << endl;

cout << tatra.getName() << " быстрее чем " << kamaz.getName() << "!!!" << endl;

}

else {

kamaz.display();

tatra.display();

cout << endl;

cout << "Их скорости одинаковые!!!" << endl;

}

return 0;

}

Теперь класс Tatra дружественный классу Kamaz, что позволяет Kamaz иметь доступ к приватным членам Tatra.