**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ**

**КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

**КЫРГЫЗСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. И. Раззакова**

**ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

Кафедра: **Программное обеспечение компьютерных систем**

Дисциплина «**Объектно-Ориентированное Программирование**»

Отчет

Лабораторная работа №4

Выполнил: студент группы ПИ-2-21

Сатыбалдыев Нурислам

Проверил: Мусабаев Э. Б.

**Бишкек 2024**

**Задания:**

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.

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

#include <iostream>

#include <iomanip>

#include <string>

class Tatra;

class Kamaz {

private:

int speed;

std::string name;

public:

Kamaz(int spd, const std::string& nm) : speed(spd), name(nm) {}

void display() const {

std::cout << std::setw(10) << name << ": " << std::setw(3) << speed << " км/ч";

}

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

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

friend class Tatra;

};

class Tatra {

private:

int speed;

std::string name;

public:

Tatra(int spd, const std::string& nm) : speed(spd), name(nm) {}

void display() const {

std::cout << std::setw(10) << name << ": " << std::setw(3) << speed << " км/ч";

}

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

};

int FrCreater(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, "Ru");

Kamaz kamaz(120, "КАМАЗ");

Tatra tatra(110, "Tatra");

std::cout << std::setw(20) << "Участок трассы" << std::setw(20) << "Победитель" << std::endl;

std::cout << "-----------------------------------------" << std::endl;

std::cout << std::setw(20) << "Равнина: ";

int resultPlain = FrCreater(kamaz, tatra);

if (resultPlain == 1)

std::cout << std::setw(20) << "КАМАЗ";

else if (resultPlain == -1)

std::cout << std::setw(20) << "Tatra";

else

std::cout << std::setw(20) << "Ничья";

std::cout << std::endl;

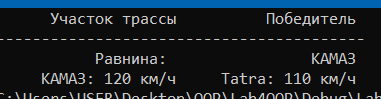
// Повторяем для других участков трассы: горы и пустыня

kamaz.display();

tatra.display();

return 0;

}

****