Министерство образования и науки Кыргызской Республики

Кыргызский государственный технический университет

им. И.Раззакова

Факультет информационных технологий

Кафедра «Программное обеспечение компьютерных систем»

Направление: 710400 «Программная инженерия»

Дисциплина: «Объектно – ориентированное программирование»

**ОТЧЕТ**

По лабораторной работе №4.

Тема: «Дружественные функции и дружественные классы»

                                                                       Выполнила: студент группы

ПИ(б)-2-21 Марат кызы Жылдыз

Проверил: Мусабаев Э. Б.

**Бишкек – 2024**

**Лабораторная работа №4.**

**Тема: Дружественные функции и дружественные классы.**

**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.

**Код программы:**

#include<iostream>

#include<Windows.h>

#include<iomanip>

#include<ctime>

#include<cstdlib>

using namespace std;

class Tatra;

class Kamaz

{

private:

string nameK;

float speedK;

friend class Tatra;

public:

Kamaz() :nameK(""), speedK(0) {}

~Kamaz() {}

friend int FrCreater(Kamaz objk, Tatra objT);

void input() {

cout << "Введите наименование грузовика: ";

cin >> nameK;

cout << "Введите скорость (км/час): ";

cin >> speedK;

cout << endl;

}

};

class Tatra

{

private:

string nameT;

float speedT;

public:

Tatra() : nameT(""), speedT(0) {}

~Tatra() {}

friend int FrCreater(Kamaz oblK, Tatra oblT);

void input() {

cout << "Введите наименование грузовика: ";

cin >> nameT;

cout << "Введите скорость (км/час): ";

cin >> speedT;

cout << endl;

}

void Show(Kamaz K) {

cout << "\tИнформация о грузовике TATRA" << endl;

cout << setw(5) << "Наименование грузовика" << setw(30) << "Скорость грузовика" << endl;

cout << setw(5) << nameT << setw(30) << speedT << "км/час" << endl;

cout << "\tИнформация о грузовике KAMAZ" << endl;

cout << setw(5) << "Наименование грузовика" << setw(30) << "Скорость грузовика" << endl;

cout << setw(5) << K.nameK << setw(30) << K.speedK << "км/час" << endl;

}

};

int FrCreater(Kamaz oblK, Tatra oblT)

{

if (oblK.speedK > oblT.speedT) {

cout << "Победитель марафона грузовик KAMAZ: " << oblK.nameK << endl;

return 1;

}

else if (oblK.speedK == oblT.speedT) {

cout << "Победителя нет!" << endl;

return 0;

}

else {

cout << "Победитель марафона грузовик TATRA: " << oblT.nameT << endl;

return -1;

}

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(LC\_ALL, "RUS");

int val = 0;

string pr;

Kamaz Km;

Tatra Tr;

cout << "Проходимости: " << "равнины, горы, пустыня" << endl;

cout << "Введите проходимость: " << endl;

cin >> pr;

while (val != 4)

{

system("CLS");

cout << "\nМеню" << endl;

cout << "1) Ввод" << "\n2) Вывод" << "\n3) Вычислить побидителя" << endl;

cout << endl << "Выберите действие" << endl;

cin >> val;

if (val == 1) {

system("CLS");

cout << "\tГрузовик KAMAZ" << endl;

Km.input();

cout << endl << "\tГрузовик TATRA" << endl;

Tr.input();

cout << "Данные записаны!" << endl;

system("pause");

system("CLS");

}

else if (val == 2) {

system("CLS");

Tr.Show(Km);

system("pause");

system("CLS");

}

else if (val == 3) {

system("CLS");

cout << "\tРезультаты ралли-марафона по проходимости " << " ' " << pr << " ' " << endl;

cout << endl << FrCreater(Km, Tr) << endl;

system("pause");

system("CLS");

}

}

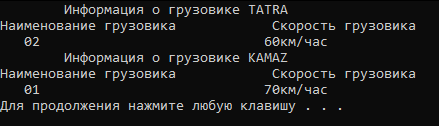
}

**Результат программы:**

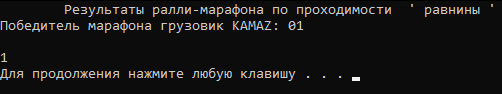
**Рис.1. Выбор проходимости.**

****

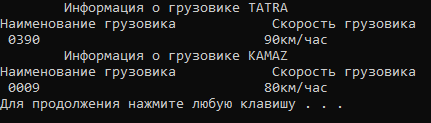
**Рис.2. Информация о грузовиках (равнины)**



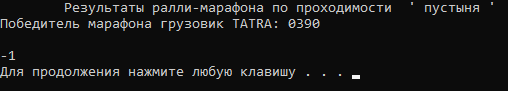
**Рис.3. Победитель (равнины)**



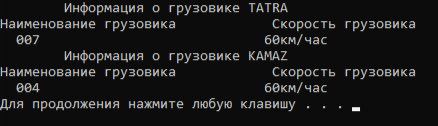
**Рис.4. Информация о грузовиках (пустыня)**



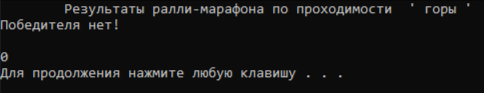
**Рис.5. Победитель (пустыни)**



**Рис.6. Информация о грузовиках (горы)**



**Рис.7. Победитель (горы)**



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

**Код программы (объедены задания 2 и 3):**

#include<iostream>

#include<string>

#include<iomanip>

#include<Windows.h>

double aveST; //Средняя скорость грузовика Татра

double aveSK; //Средняя скорость грузовика Камаз

int TatraScore; //Очки грузовика Татра

int KamazScore; //Очки грузовика Камаз

using namespace std;

class Tatra;//Предварительное объявление

class Kamaz //Класс Камаз

{

string nameK; //Номер

double speedDesert; //Скорость на пустыне

double speedMountain; //Скорость в горах

double speedPlain; //Скорость в равнинах

public:

//Метод Камаз

Kamaz(string n, double sD, double sM, double sP)

{

nameK = n;

speedDesert = sD;

speedMountain = sM;

speedPlain = sP;

};

//Дружественные функции

friend int FrCreator(Kamaz s1, Tatra s2);

friend void AveSpeed(Kamaz s1, Tatra s2);

friend void ScoreCount(Kamaz s1, Tatra s2);

//Возвращение значения

string showName()

{

return nameK;

}

double showsD()

{

return speedDesert;

}

double showsM()

{

return speedMountain;

}

double showsP()

{

return speedPlain;

}

};

class Tatra

{

string nameT;

double speedDesert;

double speedMountain;

double speedPlain;

public:

Tatra(string n, double sD, double sM, double sP)

{

nameT = n;

speedDesert = sD;

speedMountain = sM;

speedPlain = sP;

};

friend int FrCreator(Kamaz s1, Tatra s2);

friend void AveSpeed(Kamaz s1, Tatra s2);

friend void ScoreCount(Kamaz s1, Tatra s2);

string showName()

{

return nameT;

}

double showsD()

{

return speedDesert;

}

double showsM()

{

return speedMountain;

}

double showsP()

{

return speedPlain;

}

};

//Функции

int FrCreator(Kamaz s1, Tatra s2)

{

int c = 0;

if (s1.speedDesert > s2.speedDesert) c++;

else if (s1.speedDesert == s2.speedDesert) c = c;

else c--;

if (s1.speedMountain > s2.speedMountain) c++;

else if (s1.speedMountain == s2.speedMountain) c = c;

else c--;

if (s1.speedPlain > s2.speedPlain) c++;

else if (s1.speedPlain == s2.speedPlain) c = c;

else c--;

return c;

}

void AveSpeed(Kamaz s1, Tatra s2)

{

double a = s1.showsD(), b = s1.showsM(), c = s1.showsP();

aveSK = (a + b + c) / 3;

a = s2.showsD(), b = s2.showsM(), c = s2.showsP();

aveST = (a + b + c) / 3;

}

void ScoreCount(Kamaz s1, Tatra s2)

{

if (s1.speedDesert > s2.speedDesert)

KamazScore++;

else if (s1.speedDesert == s2.speedDesert)

{

KamazScore++;

TatraScore++;

}

else

TatraScore++;

if (s1.speedMountain > s2.speedMountain)

KamazScore++;

else if (s1.speedMountain == s2.speedMountain)

{

KamazScore++;

TatraScore++;

}

else

TatraScore++;

if (s1.speedPlain > s2.speedPlain)

KamazScore++;

else if (s1.speedPlain == s2.speedPlain)

{

KamazScore++;

TatraScore++;

}

else

TatraScore++;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

string Rnumber;

double sD, sM, sP;

cout << "Меряем скорость грузовика \*KAMAZ\*" << endl;

cout << "\nKamaz номер: ";

cin >> Rnumber;

cout << "\nСкорость Kamaz'а на песке: ";

cin >> sD;

cout << "\nСкорость Kamaz'a на горах: ";

cin >> sM;

cout << "\nСкорость Kamaz'а на равнинах: ";

cin >> sP;

Kamaz Kam(Rnumber, sD, sM, sP);

cout << "\n\nМеряем скорость грузовика \*TATRA\*" << endl;

cout << "\nTATRA номер: ";

cin >> Rnumber;

cout << "\nСкорость Tatra на песке: ";

cin >> sD;

cout << "\nСкорость Tatra на горах: ";

cin >> sM;

cout << "\nСкорость Tatra на равнинах: ";

cin >> sP;

Tatra Tat(Rnumber, sD, sM, sP);

AveSpeed(Kam, Tat);

ScoreCount(Kam, Tat);

cout << endl;

cout << " \*----------" << "\*-------" << "\*----------" << "\*----------" << "\*----------" << "\*-------------------" << "\*----------" << "\*------------" << "\* \n";

cout << " |" << setw(9) << "Грузовик" << " |" << setw(6) << "Номер" << " |" << setw(9) << "Пустыня" << " |" << setw(9) << "Горы" << " |" << setw(9) << "Равнины" << " |" << setw(18) << "Средняя скорость" << " |" << setw(9) << "Очки" << " |" << setw(9) << "Вывод |" << endl;

cout << " \*----------" << "\*-------" << "\*----------" << "\*----------" << "\*----------" << "\*-------------------" << "\*----------" << "\*------------" << "\* \n";

cout << " |" << setw(9) << "Kamaz" << " |" << setw(6) << Kam.showName() << " |" << setw(9) << Kam.showsD() << " |" << setw(9) << Kam.showsM() << " |" << setw(9) << Kam.showsP() << " |" << setw(18) << aveSK << " |" << setw(9) << KamazScore << " |" << setw(9);

if (FrCreator(Kam, Tat) > 0) cout << "Победитель " << "| \n";

else if (FrCreator(Kam, Tat) < 0) cout << "Проигравший " << "| \n";

else cout << "Никто " << "| \n";

cout << " |" << setw(9) << "TATRA" << " |" << setw(6) << Tat.showName() << " |" << setw(9) << Tat.showsD() << " |" << setw(9) << Tat.showsM() << " |" << setw(9) << Tat.showsP() << " |" << setw(18) << aveST << " |" << setw(9) << TatraScore << " |" << setw(9);

if (FrCreator(Kam, Tat) < 0) cout << "Победитель " << "| \n";

else if (FrCreator(Kam, Tat) > 0) cout << "Проигравший " << "| \n";

else cout << "Никто " << "| \n";

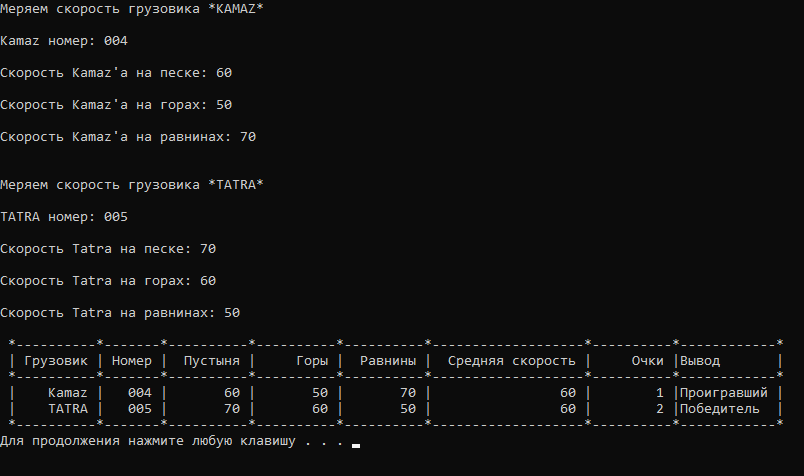
cout << " \*----------" << "\*-------" << "\*----------" << "\*----------" << "\*----------" << "\*-------------------" << "\*----------" << "\*------------" << "\* \n";

system("pause");

return 0;

}

**Результат программы:**

****