***Министерство образования Республики Беларусь***

***Учреждение образования***

***«Брестский государственный технический университет»***

***Кафедра ИИТ***

**Лабораторная работа №**

**По дисциплине ППвИС за III семестр**

**Тема: “Сравнение скоростей”**

**Выполнил:**

Студент группы ИИ-15 (1)

2-го курса

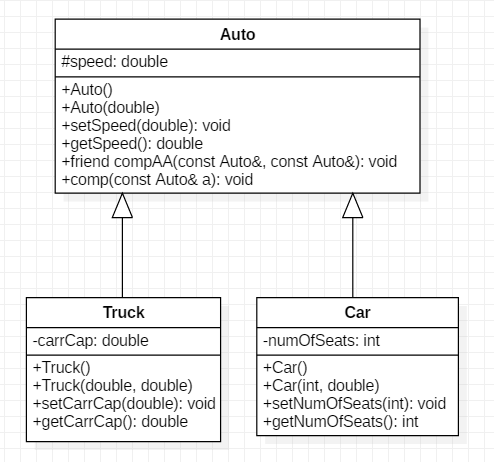
Волк И. А.

**Проверил:**

Муравьев Г. Л.

Брест 2018

UML:



Листинг программы:

#include <iostream>

class Auto

{

protected:

double speed;

public:

// Constructors/Destructor

Auto();

Auto(double speed);

// Methods

void setSpeed(double speed);

double getSpeed();

friend void compAA(const Auto&, const Auto&);

void comp(const Auto& a);

};

class Truck : public Auto

{

private:

// carring capacity

double carrCap;

public:

// Constructors/Destructor

Truck();

Truck(double carrCap, double speed);

// Methods

void setCarrCap(double carrCap);

double getCarrCap();

};

class Car : public Auto

{

private:

int numOfSeats;

public:

// Constructors/Destructor

Car();

Car(int numOfSeats, double speed);

// Methods

void setNumOfSeats(int numOfSeats);

int getNumOfSeats();

};

void compAA(const Auto& auto1, const Auto& auto2);

void compCT(Car& car, Truck& truck);

int main()

{

Car car(4, 100.0);

Truck truck(100.0, 50.0);

compAA(car, truck);

compCT(car, truck);

car.comp(truck);

system("pause");

return 0;

}

Auto::Auto()

{

this->speed = 0;

}

Auto::Auto(double speed)

{

try

{

if (speed < 0)

throw 1;

this->speed = speed;

}

catch (int e)

{

std::cout << "Auto::Auto(double)\t\tError - " << e << std::endl;

}

}

void Auto::setSpeed(double speed)

{

try

{

if (speed < 0)

throw 1;

this->speed = speed;

}

catch (int e)

{

std::cout << "Auto::setSpeed(double)\t\tError - " << e << std::endl;

}

}

double Auto::getSpeed()

{

return this->speed;

}

void Auto::comp(const Auto& a)

{

if (this->speed > a.speed)

std::cout << "I'm faster!" << std::endl;

else if (this->speed == a.speed)

std::cout << "We are equal!" << std::endl;

else

std::cout << "I'm Slower!" << std::endl;

}

Truck::Truck()

{

this->speed = 0;

this->carrCap = 0;

}

Truck::Truck(double carrCap, double speed)

{

try

{

if (speed < 0)

throw 1;

this->speed = speed;

if (carrCap < 0)

throw 2;

this->carrCap = carrCap;

}

catch (int e)

{

std::cout << "Truck::Truck(double, double)\t\tError - " << e << std::endl;

}

}

void Truck::setCarrCap(double carrCap)

{

try

{

if (carrCap < 0)

throw 1;

this->carrCap = carrCap;

}

catch (int e)

{

std::cout << "Truck::setCarrCap(double)\t\tError - " << e << std::endl;

}

}

double Truck::getCarrCap()

{

return this->carrCap;

}

Car::Car()

{

this->speed = 0;

this->numOfSeats = 0;

}

Car::Car(int numOfSeats, double speed)

{

try

{

if (speed < 0)

throw 1;

this->speed = speed;

if (numOfSeats < 0)

throw 2;

this->numOfSeats = numOfSeats;

}

catch (int e)

{

std::cout << "Car::Car(int, double)\t\tError - " << e << std::endl;

}

}

void Car::setNumOfSeats(int numOfSeats)

{

try

{

if (numOfSeats < 0)

throw 1;

this->numOfSeats = numOfSeats;

}

catch (int e)

{

std::cout << "Car::setNumOfSeats(int)\t\tError - " << e << std::endl;

}

}

int Car::getNumOfSeats()

{

return this->numOfSeats;

}

void compAA(const Auto& auto1, const Auto& auto2)

{

if (auto1.speed > auto2.speed)

std::cout << "Auto1 is faster!" << std::endl;

else if (auto1.speed == auto2.speed)

std::cout << "Auto1 and auto2 have the same speed!" << std::endl;

else

std::cout << "Auto2 is faster!" << std::endl;

}

void compCT(Car& car, Truck& truck)

{

int truckSpeed = truck.getSpeed();

int carSpeed = car.getSpeed();

if (truckSpeed > carSpeed)

std::cout << "Truck is faster!" << std::endl;

else if (truckSpeed == carSpeed)

std::cout << "Tuck and car have the same speed!" << std::endl;

else

std::cout << "Car is faster!" << std::endl;

}

Результат выполнения:

