Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №6**

По дисциплине «Проектирование программного обеспечения интеллектуальных систем»

Тема: «Пространства имен. Исключения»

**Выполнил:**

Студент 2 курса

Группы ИИ-21

Литвинюк Т. В.

**Проверил:**

Монтик Н. С.

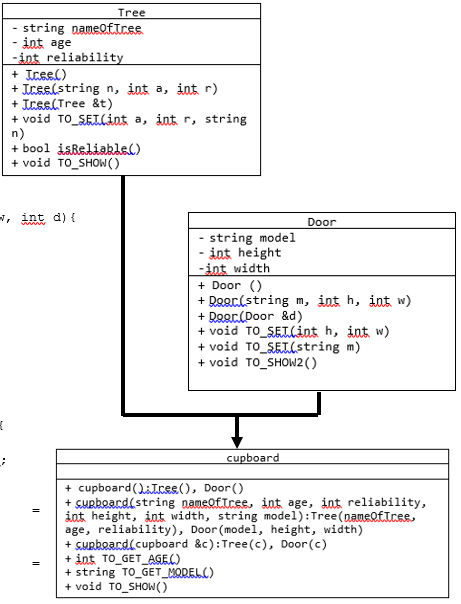
Брест 2022

**Цель:** научиться использовать пространство имен.

**Ход работы:**

#include <iostream>

using namespace std;

namespace TreeNS{

class Tree{

public:

string nameOfTree;

int age, reliability;

Tree(){

nameOfTree = "";

age = 0;

reliability = 0;

}

Tree(string n, int a, int r){

nameOfTree = n;

age = a;

reliability = r;

}

Tree(Tree &t){

nameOfTree = t.nameOfTree;

age = t.age;

reliability = t.reliability;

}

void TO\_SET(int a, int r, string n){

age = a; reliability = r; nameOfTree = n;

}

bool isReliable(){

if(reliability > 50)

return true;

return false;

}

void TO\_SHOW(){

cout << "nameOfTree: " << nameOfTree << endl;

cout << "age: " << age << "\nreliability: " << reliability << endl;

}

};

}

namespace DoorNS{

class Door{

public:

int height, width; string model;

Door(){

model = "";

height = 0;

width = 0;

}

Door(string m, int h, int w){

model = m;

height = h;

width = w;

}

Door(Door &d){

this->model = d.model;

this->height = d.height;

this->width = d.width;

}

void TO\_SET(int h, int w){

height = h; width = w;

}

void TO\_SET(string m){

model = m;

}

void TO\_SHOW2(){

cout << "Furniture: Door" << endl;

cout << "Model: " << model << endl;

cout << "Height: " << height << endl;

cout << "Width: " << width << endl;

}

};

}

namespace CupboardNS{

class cupboard : public DoorNS::Door, public TreeNS::Tree{

public:

cupboard():Tree(), Door(){}

cupboard(string nameOfTree, int age, int reliability,

int height, int width, string model):Tree(nameOfTree, age, reliability), Door(model, height, width){}

cupboard(cupboard &c):Tree(c), Door(c){};

int TO\_GET\_AGE(){

return Tree::age;

}

string TO\_GET\_MODEL(){

return Door::model;

}

int difference(int age2){

try{

if(age2 < 0)

throw "Age cannot be negative";

return age2 - age;

} catch (const char\* msg) {

cerr << msg << endl;

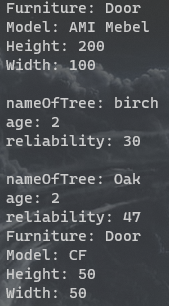
}

}

void TO\_SHOW(){

Tree::TO\_SHOW();

Door::TO\_SHOW2();

 }

};

}

int main(){

using namespace DoorNS;

using namespace TreeNS;

using namespace CupboardNS;

Door d("AMI Mebel", 200, 100);

d.TO\_SHOW2();

cout << endl;

Tree t("birch", 2, 30);

t.TO\_SHOW();

cout << endl;

cupboard cupboard1("Oak", 2, 47, 50, 50, "CF");

cupboard1.TO\_SHOW();

}

**Вывод:** в ходе лабораторной работы я научился использовать ссылочный тип в классах.