Lab 5.3.11 Inheritance basics: part 1

Objectives

Familiarize the student with:

- inheritance syntax and operation;
- sharing functionality between objects using inheritance.

Scenario

Farm owners have to know how much water they need in order to provide enough for their animals.

All farm animals need water, so they share a common trait.

Let's model this relationship with C++ classes.

You have no particular task in this lab.

Play around with the code below, add some print statements in the constructors to see when they're called and with what arguments.

Try to modify the member of the base class from the derived class and see if it's possible. Then think of several ways you can modify the code to make it possible.

```cpp
#include <iostream>

using namespace std;

class FarmAnimal{
public:
  FarmAnimal(double water_consumption);
  double getWaterConsumption();
  // ...
private:
  double water_consumption;
};

FarmAnimal::FarmAnimal(double water_consumption) {
  this->water_consumption = water_consumption;
}

double FarmAnimal::getWaterConsumption() {
  return water_consumption;
}


class DummyAnimal : public FarmAnimal{
public:
  DummyAnimal();
};

DummyAnimal::DummyAnimal() : FarmAnimal(10.0)
{

}


class DoublingAnimal : public FarmAnimal
{
public:
  DoublingAnimal(double given_water_consumption);
};
```

```cpp
DoublingAnimal::DoublingAnimal(double given_water_consumption)
: FarmAnimal(2 * given_water_consumption) {

}

void printConsumption(FarmAnimal "animal)
{
  cout << "This animal consumes " << animal.getWaterConsumption()
       << " liters of water" << endl;
}

int main()
{
  FarmAnimal animalA(5);
  DummyAnimal animalB;
  DoublingAnimal animalC(21);

  cout << "animalA consumes " << animalA.getWaterConsumption()
       << " liters of water." << endl;
  cout << "What about the others?" << endl;

  printConsumption(animalB);
  printConsumption(animalC);

  return 0;
}
```