

# OO-programming Exercise 2

Due Date and Time: Nov 3, 2009, 23:59

---

You should work out the exercises independently.

Send the sensibly documented program files to gundolf.haase@uni-graz.at or bring them with you (Files for Example. 5: `bsp_5.cpp`, `bsp_5_fkt.cpp`, `bsp_5_fkt.hpp`, etc.) on Laptop or USB-stick

---

2. Write a class hierarchy of animals at a farm and integrate your class `cow` (or whatever name you use) into it. Include also working animals as *dog*, *sheep* and *chicken* into the hierarchy.

What are common properties (basis class) and what are differentiating properties (derived classes) of the working animals at the farm?

Each class should contain a method that calculates the animal's value and the name of the animal should be accessible.

3. Design and implement your class hierarchy such that the basis class (let's name it `animal`) is an abstract class.
4. Test your implementation by creating an array of `animals` (pointer or references) and use polymorphism in a loop when accessing the element arrays.
5. Derive subclasses of `cow` as *milker*, *ox*, *bull* and integrate them into your class hierarchy. Test them similarly to task 4.  
You might even try dynamic binding.