

Object-Oriented Programming (List 8)

Deadline: 17.05.2011

This exercise is about constructing an artificial example with inheritance.

1. Create a simple class hierarchy, consisting of classes `aaaa`, `bbbb1`, `bbbb2`, `cccc11`, `cccc12`. It must be the case that `bbbb1`, `bbbb2` inherit from `aaaa`, and `cccc11`, `cccc12` inherit from `bbbb1`.

Assume that class `aaaa` has one field `unsigned int x1`. Assume that classes `bbbb1` and `bbbb2` have an additional field `unsigned int x2`. Assume that classes `cccc11`, `cccc12` have a third, additional field `unsigned int x3`.

Each of the classes must have a suitable constructor with a suitable arity.

(There will probably be difficulties with initialization of super classes, we will see during exercise.)

It is OK to put all classes in a single file.

2. Give each of the classes a suitable `print(std::ostream&) const` method. Implement a function `std::ostream& operator << ()` that prints its second argument in a suitable way.
3. Define a member function `unsigned int sum() const` that sums all members of a `aaaa`, `bbbb1`, `bbbb2`, `cccc11`, `cccc12`.
4. Test the previous tasks with

```
int main( int argc, char* argv [ ] )
{
    aaaa a(4);
    cccc11 c( 1,2,3);
    bbbb2 b(4,5);

    std::cout << a << " " << b << " " << c << "\n";
    std::cout << a. sum( ) + b. sum( ) + c. sum( ) << "\n";
}
```

(or something similar.)

(See the rules for showing code on the course homepage. Default value of a task (when shown on time) is 3 points.)