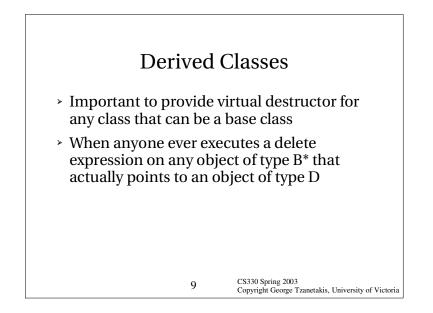
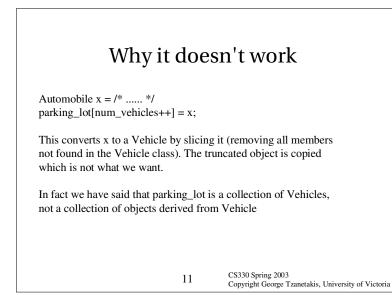


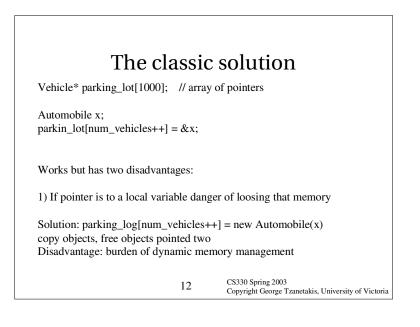
7

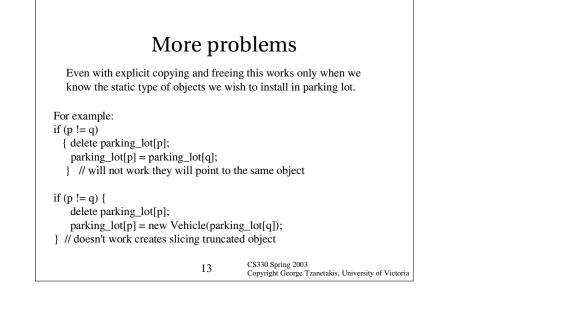
Copyright George Tzanetakis, University of Victoria

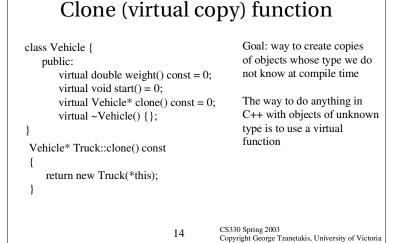


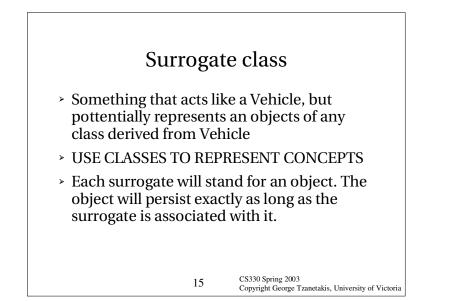
Surrogate classes		
<pre>class Vehicle {     public:         virtual double weight() con         virtual void start() = 0; };</pre>	nst = 0;	
class RoadVehicle: public Vehicle {} class Automobile : public RoadVehicle We want to have a container of Vehicles of different kinds : Vehicle parking_lot[1000]; // this doesn't work		
1	0	CS330 Spring 2003 Copyright George Tzanetakis, University of Victoria

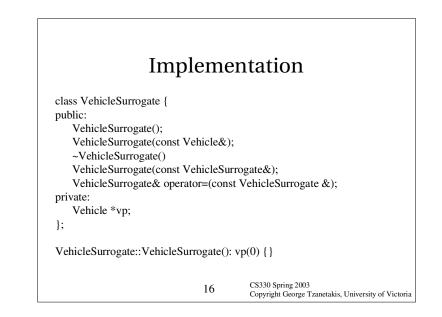


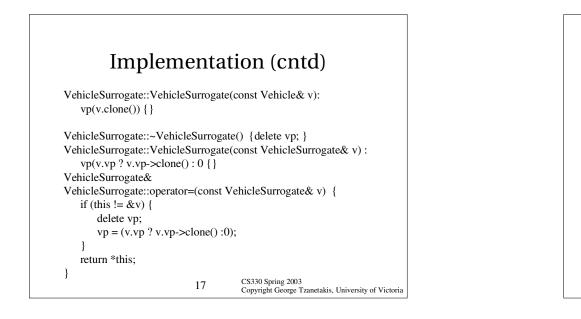


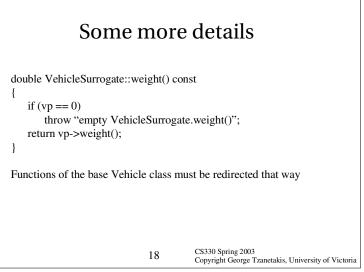












## What have we achieved ?

VehicleSurrogate parking\_lot[1000]; Automobile x; parking\_lot[num\_vehicles++] = x;

same thing as: parking\_lot[num\_vehicles++] = VehicleSurrogate(x);

When parking lot deleted all copies are deleted.

Some problems: copies can be costly for large objects (use counts are a possible solution)

19

CS330 Spring 2003

Copyright George Tzanetakis, University of Victoria