

APTCV: exam preparation

Gyuri Dorko

July 18, 2007

•	C++ General	Lectures	0,1
	What is the difference between C and C++?		
	How do you link (mix) a C++ program to C functions?		
	Explain the following code: <code>delete this;</code>		
	What is the difference between a function definition and declaration?		
	What is the difference between a pointer and a reference?		
•	C++ Definitions	Lectures	1,2,3,9
	What is a <i>side-effect</i> ?		
	What are the differences between a C++ struct and C++ class?		
	Explain the difference between the following two lines. <code>MySpace::Image img; // line (1)</code> <code>MySpace::Image img(); // line (2)</code>		
	What is the <i>initialization list</i> of constructors? Show an example.		
	What is a POD type? How is the initialization of POD objects differ from non-POD objects?		
	Explain the following code: <code>sizeof(A);</code>		
	Define a function called <code>max</code> with two integer parameters, and a return value of integer. Define a function pointer pointing to the <code>max</code> function.		
	What does the following variable definition promises: <code>const int* p = &i;</code>		
	What is a constant member function?		
	Why const correctness is important?		
	What is a mutable member?		
	What problem does the namespace feature solve?		
	What is argument dependent name lookup (Koenig lookup)?		
	What is an unnamed namespace? When should one use it?		
	What does the following line do: <code>using namespace std;</code>		
	What are the four special member functions?		
	When does the compiler provide default implementation for the special member functions?		
	What is a conversion constructor?		
	What is an explicit constructor?		
	What is the difference between a copy constructor and an overloaded assignment operator?		

•	Memory management	Lectures	3,4,6
	What are the five different memory areas? What are their properties? Show a variable definition for each.		
	What does the <i>placement new</i> operator do? Show an example call. Why is it useful?		
	When do a class need a virtual destructor?		
	What is a <i>memory leak</i> , how can be detected?		
	What are smart pointers?		
	Show an example, when a temporary is created.		
	What is the "static fiasco"? Describe a possible solution for such problem.		
•	Standard Template Library	Lectures	3,4,5,8
	What is the Standard Template Library?		
	What is <code>std::auto_ptr<T></code> ?		
	What are the most important properties of <code>std::vector</code> ?		
	What are the most important properties of <code>std::map</code> ?		
	How to remove elements from a standard container, like list, map, or vector?		
	What is the erase-remove idiom?		
	What does <code>vector::reserve()</code> do?		
	What is the advantage of using function object over function pointers? Are they always interchangeable?		
	What is a binder in STL. Tell an example.		
	What is an adaptor in STL. Tell an example.		
•	Programming/Design Techniques	Lectures	2,3,4,6,9
	What is a Singleton?		
	What is a Phoenix Singleton?		
	How would you implement most efficiently a a callback-like interface if the callee known at compile time?		
	What is a <i>named constructor</i> idiom?		
	Explain the <i>Return Value Optimization</i> .		
	What is function inlining? How is it realized in C++?		
	When can/cannot a function be inlined?		
	In which ways (why) inlining may lead better performance?		
	Tell a few typical examples for compiler optimizations.		
	What is a policy based design?		

•	Templates	Lectures	8,9,10
	What is a member template function?		
	What is the difference between implicit and explicit instantiation?		
	What is partial specialization? Show an example.		
	What are non-type template arguments? Show an example.		
	What do we call <i>concepts</i> in generic programming?		
	What are <i>traits</i> in generic programming? Tell an example for a <i>type trait</i> .		
	What is an archetype and how is it related to concepts?		
	What is a Curiously Reoccurring Template Pattern (CRTTP)? Tell an example application of this pattern.		
	What is the Barton-Nackman trick?		
•	C++ & Computer Vision	Lectures	1,3,5,7
	What do we call the Region of Interest (ROI) or Subimage in a case of image representation. How would you implement it?		
	What is the difference between interleaved and non-interleaved image layout?		
	When up-scaling (magifing) an image what techniques are used to fill the "missing" pixels?		
	What are the steps of downscaling an image?		
	What is the difference between local and global image representation?		
	When can we call a local image descriptor rotation invariant? Describe a rotation invariant descriptor.		
	What is an interest point detector?		
	What is the Scale Invariant Feature Transform (SIFT)?		
	What is clustering for?		
	What type of clustering methods do you know?		
	Describe the memory requirements of a chosen clustering method.		

Expect some programming examples, like making a small change in a given program, explaining what a certain piece of C++ code does, writing a small example program. Look at the quizzes (available on the web site) and the examples from the the slides! You will also be asked about your code.