

Re: Storing a tree in a std::list<>

Source: <http://coding.derkeiler.com/Archive/C/ CPP/comp.lang.cpp/2004-06/0835.html>

From: Alan Johnson (alanwj_at_mailandnews.com)

Date: 06/06/04

Date: Sat, 05 Jun 2004 19:45:58 -0500

Dave wrote:

> *Hello all,*

>

> *After perusing the Standard, I believe it is true to say that once you*

> *insert an element into a std::list<>, its location in memory never changes.*

> *This makes a std::list<> ideal for storing vertices of an arbitrary n-ary*

> *tree where a vertex contain pointers to its parent / children. These parent*

> */ child vertices need to stay put if we've got pointers to them somewhere!*

>

> *Am I correct in my assertion?*

>

> *Thanks,*

> *Dave*

>

>

While that probably is true for almost any implementation, I don't think that the standard actually requires it. What it does require is that adding/removing elements (as well as most other operations) do not invalidate any iterators to elements of the list.

Alan