

```

datetime.h row 1733
inline wxLongLong wxDateTime::GetValue() const
{
>>    wxASSERT_MSG( IsValid(), wxT("invalid wxDateTime"));

    return m_time;
}

```

```

and then at row 858
inline bool operator>(const wxDateTime& dt) const
{
>>    wxASSERT_MSG( IsValid() && dt.IsValid(), wxT("invalid wxDateTime") );
    return GetValue() > dt.GetValue();
}

```

```

Output:
Call stack:
[00] wxDateTime::operator<=
c:\wxwidgets-3.0.2\include\wx\datetime.h:853
[01] ActiveTrack::DoExtendDaily                               c:\builds\ocpn\opencpn\src\track.cpp:280
[02] MyFrame::TrackOff
c:\builds\ocpn\opencpn\src\chart1.cpp:4516
[03] MyFrame::OnCloseWindow
c:\builds\ocpn\opencpn\src\chart1.cpp:3278
[04] wxAppConsoleBase::HandleEvent
c:\wxwidgets-3.0.2\src\common\appbase.cpp:612
[05] wxAppConsoleBase::CallEventHandler
c:\wxwidgets-3.0.2\src\common\appbase.cpp:624
[06] wxEvtHandler::ProcessEventIfMatchesId
c:\wxwidgets-3.0.2\src\common\event.cpp:1394
[07] wxEventHashTable::HandleEvent
c:\wxwidgets-3.0.2\src\common\event.cpp:998
[08] wxEvtHandler::TryHereOnly
c:\wxwidgets-3.0.2\src\common\event.cpp:1589
[09] wxEvtHandler::TryBeforeAndHere
c:\wxwidgets-3.0.2\include\wx\event.h:3671
[10] wxEvtHandler::ProcessEvent
c:\wxwidgets-3.0.2\src\common\event.cpp:1491
[11] wxEvtHandler::DoTryChain
c:\wxwidgets-3.0.2\src\common\event.cpp:1554
[12] wxEvtHandler::ProcessEventLocally
c:\wxwidgets-3.0.2\src\common\event.cpp:1522
[13] wxEvtHandler::ProcessEvent
c:\wxwidgets-3.0.2\src\common\event.cpp:1495
[14] wxEvtHandler::SafelyProcessEvent
c:\wxwidgets-3.0.2\src\common\event.cpp:1613
[15] wxWindowBase::HandleWindowEvent
c:\wxwidgets-3.0.2\src\common\wincmn.cpp:1526
[16] wxWindowBase::Close
c:\wxwidgets-3.0.2\src\common\wincmn.cpp:589
[17] wxFrame::MSWWindowProc                                   c:\wxwidgets-3.0.2\src\msw\frame.cpp:848
[18] wxWndProc
c:\wxwidgets-3.0.2\src\msw>window.cpp:2711
[19] SetManipulationInputTarget
[20] CallWindowProcW
opencpn.exe has triggered a breakpoint.

```

```

call stack:
> wxbase30ud_vc_custom.dll!wxDateTime::GetValue() Line 1733 C++
wxbase30ud_vc_custom.dll!wxDateTime::operator<=(const wxDateTime & dt) Line 853 C++
opencpn.exe!ActiveTrack::DoExtendDaily() Line 280 C++
opencpn.exe!MyFrame::TrackOff(bool do_add_point) Line 4516 C++
opencpn.exe!MyFrame::OnCloseWindow(wxCloseEvent & event) Line 3278 C++
wxbase30ud_vc_custom.dll!wxAppConsoleBase::HandleEvent(wxEvtHandler * handler, void
(wxEvent &) * func, wxEvent & event) Line 612 C++
wxbase30ud_vc_custom.dll!wxAppConsoleBase::CallEventHandler(wxEvtHandler * handler,
wxEventFunctor & functor, wxEvent & event) Line 624 C++
wxbase30ud_vc_custom.dll!wxEvtHandler::ProcessEventIfMatchesId(const
wxEventTableEntryBase & entry, wxEvtHandler * handler, wxEvent & event) Line 1394 C++

```

```
wxbase30ud_vc_custom.dll!wxEventHashTable::HandleEvent(wxEvent & event, wxEvtHandler * self) Line 998 C++
wxbase30ud_vc_custom.dll!wxEvtHandler::TryHereOnly(wxEvent & event) Line 1589 C++
wxbase30ud_vc_custom.dll!wxEvtHandler::TryBeforeAndHere(wxEvent & event) Line 3671 C++
wxbase30ud_vc_custom.dll!wxEvtHandler::ProcessEvent(wxEvent & event) Line 1491 C++
wxbase30ud_vc_custom.dll!wxEvtHandler::DoTryChain(wxEvent & event) Line 1554 C++
wxbase30ud_vc_custom.dll!wxEvtHandler::ProcessEventLocally(wxEvent & event) Line 1522 C++
wxbase30ud_vc_custom.dll!wxEvtHandler::ProcessEvent(wxEvent & event) Line 1495 C++
wxbase30ud_vc_custom.dll!wxEvtHandler::SafelyProcessEvent(wxEvent & event) Line 1613 C++
wxmsw30ud_core_vc_custom.dll!wxWindowBase::HandleWindowEvent(wxEvent & event) Line 1526 C++
wxmsw30ud_core_vc_custom.dll!wxWindowBase::Close(bool force) Line 589 C++
wxmsw30ud_core_vc_custom.dll!wxFrame::MSWWindowProc(unsigned int message, unsigned int wParam, long lParam) Line 848 C++
wxmsw30ud_core_vc_custom.dll!wxWndProc(HWND__ * hWnd, unsigned int message, unsigned int wParam, long lParam) Line 2711 C++
[External Code]
[Frames below may be incorrect and/or missing, no symbols loaded for user32.dll]
wxmsw30ud_core_vc_custom.dll!wxWindow::MSWDefWindowProc(unsigned int nMsg, unsigned int wParam, long lParam) Line 2272 C++
wxmsw30ud_core_vc_custom.dll!wxWindow::MSWWindowProc(unsigned int message, unsigned int wParam, long lParam) Line 3651 C++
wxmsw30ud_core_vc_custom.dll!wxTopLevelWindowMSW::MSWWindowProc(unsigned int message, unsigned int wParam, long lParam) Line 467 C++
wxmsw30ud_core_vc_custom.dll!wxFrame::MSWWindowProc(unsigned int message, unsigned int wParam, long lParam) Line 887 C++
wxmsw30ud_core_vc_custom.dll!wxWndProc(HWND__ * hWnd, unsigned int message, unsigned int wParam, long lParam) Line 2711 C++
[External Code]
wxmsw30ud_core_vc_custom.dll!wxWindow::MSWDefWindowProc(unsigned int nMsg, unsigned int wParam, long lParam) Line 2272 C++
wxmsw30ud_core_vc_custom.dll!wxWindow::MSWWindowProc(unsigned int message, unsigned int wParam, long lParam) Line 3651 C++
wxmsw30ud_core_vc_custom.dll!wxTopLevelWindowMSW::MSWWindowProc(unsigned int message, unsigned int wParam, long lParam) Line 467 C++
wxmsw30ud_core_vc_custom.dll!wxFrame::MSWWindowProc(unsigned int message, unsigned int wParam, long lParam) Line 887 C++
wxmsw30ud_core_vc_custom.dll!wxWndProc(HWND__ * hWnd, unsigned int message, unsigned int wParam, long lParam) Line 2711 C++
[External Code]
wxmsw30ud_core_vc_custom.dll!wxGUIEventLoop::ProcessMessage(tagMSG * msg) Line 172 C++
wxmsw30ud_core_vc_custom.dll!wxGUIEventLoop::Dispatch() Line 232 C++
wxbase30ud_vc_custom.dll!wxEventLoopManual::ProcessEvents() Line 173 C++
wxbase30ud_vc_custom.dll!wxEventLoopManual::DoRun() Line 206 C++
wxbase30ud_vc_custom.dll!wxEventLoopBase::Run() Line 78 C++
wxbase30ud_vc_custom.dll!wxAppConsoleBase::MainLoop() Line 334 C++
wxbase30ud_vc_custom.dll!wxAppConsoleBase::OnRun() Line 260 C++
wxmsw30ud_core_vc_custom.dll!wxAppBase::OnRun() Line 305 C++
wxbase30ud_vc_custom.dll!wxEntryReal(int & argc, wchar_t * * argv) Line 495 C++
wxbase30ud_vc_custom.dll!wxEntry(int & argc, wchar_t * * argv) Line 188 C++
wxmsw30ud_core_vc_custom.dll!wxEntry(HINSTANCE__ * hInstance, HINSTANCE__ * __formal, char * __formal, int nCmdShow) Line 415 C++
opencpn.exe!WinMain(HINSTANCE__ * hInstance, HINSTANCE__ * hPrevInstance, char * __formal, int nCmdShow) Line 829 C++
[External Code]
```