

Hello,

I've encountered serious problems with the following code under windows:

```
#include <Core/Core.h>
#include <occi.h>
```

```
CONSOLE_APP_MAIN
```

```
{
    std::string rs;
    {
        oracle::occi::Environment *env =
oracle::occi::Environment::createEnvironment(oracle::occi::Environment::DEFAULT);
        oracle::occi::Connection* con = env->createConnection("Scott", "Tiger",

"(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.1.1)(POR
T=1521)))(CONNECT_DATA=(SERVICE_NAME=test.upp)))");
        oracle::occi::Statement* stmt = con->createStatement("SELECT 'ABC' FROM DUAL");
        oracle::occi::ResultSet* rset = stmt->executeQuery();
        if (rset->next()) {
            rs = rset->getString(1);
        } // LEAVING THIS SCOPE LEADS TO ERROR!
        stmt->closeResultSet(rset);
        env->terminateConnection(con);
        oracle::occi::Environment::terminateEnvironment(env);
    }
    if (rs.length()) {

    }
}
```

ERROR (running in debug):

This test-scenario only fails in debug actually; I haven't found a simple version for release mode until now.

What am I doing wrong here?

Why tries Ultimate to free memory, that the occi library created using the msvc crt??

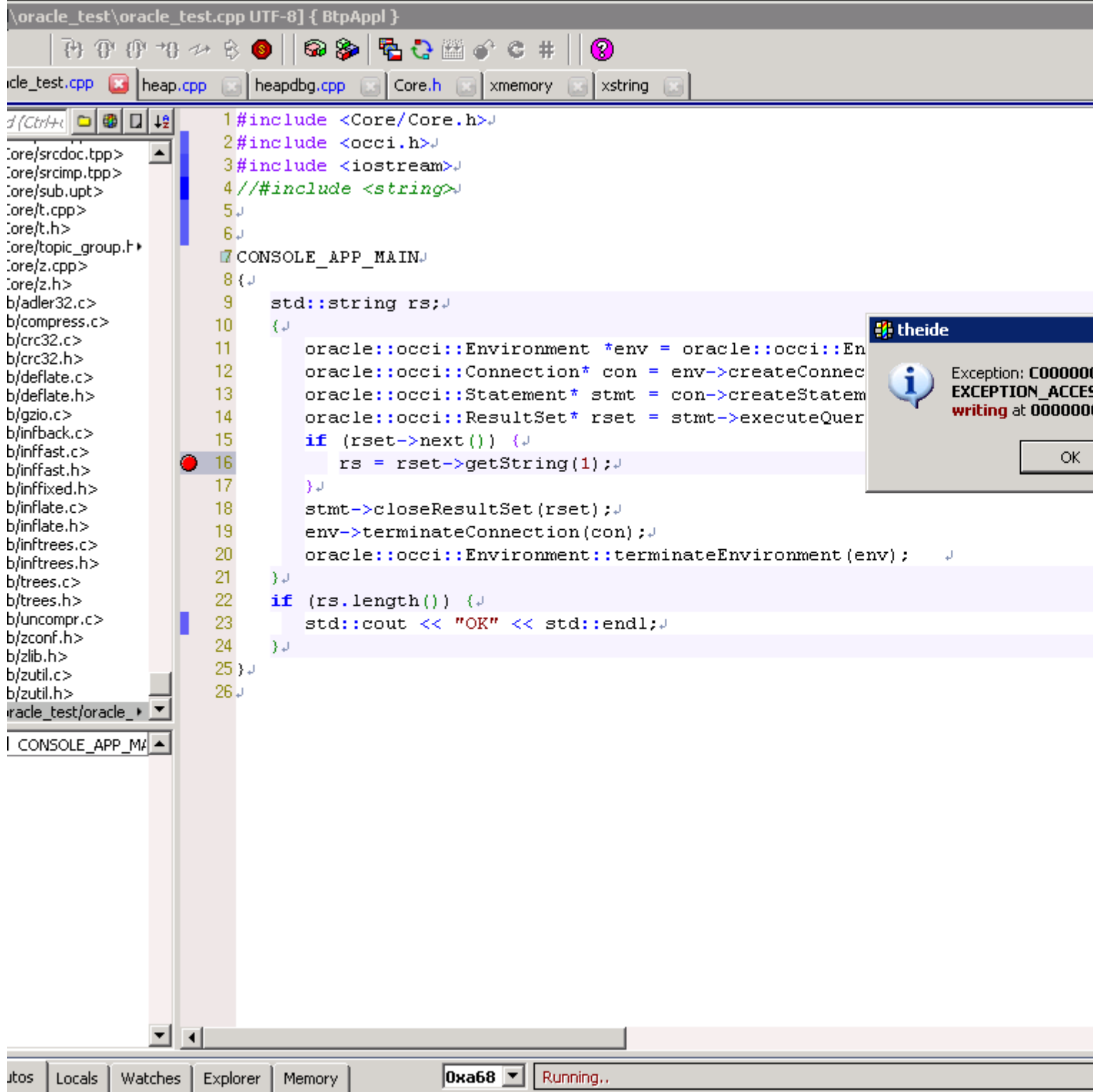
Am I missing something? Can you give me some hints?

Kind regards
wqcmaster

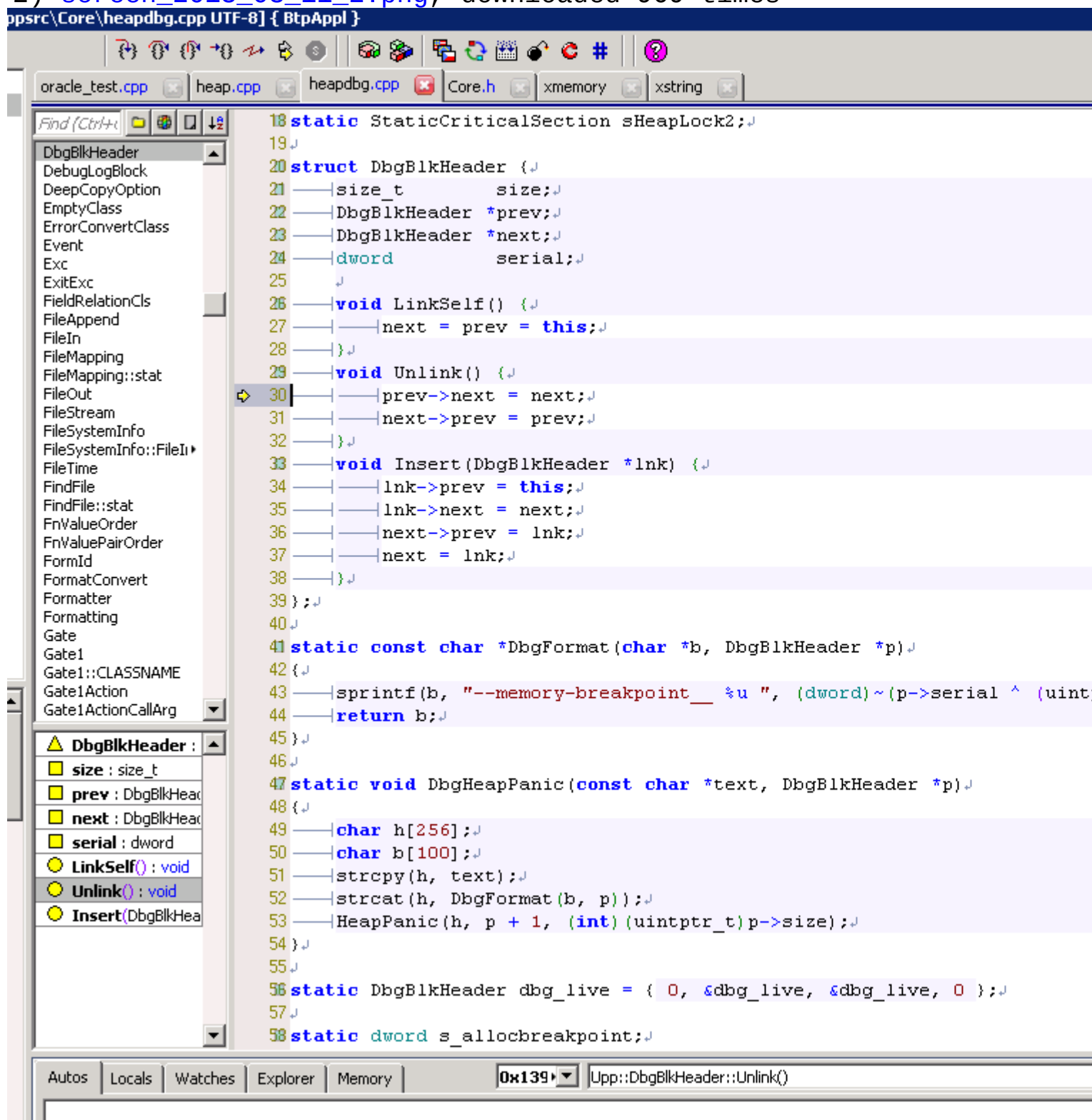
P.S.: Using UPP version 5485; Visual Studio 2010; Oracle OCCI Library 11.2 with VC10

File Attachments

1) [screen_2013_03_11_1.png](#), downloaded 1003 times



2) [screen_2013_03_11_2.png](#), downloaded 969 times



```
18 static StaticCriticalSection sHeapLock2;
19
20 struct DbgBlkHeader {
21     size_t      size;
22     DbgBlkHeader *prev;
23     DbgBlkHeader *next;
24     dword      serial;
25
26     void LinkSelf() {
27         next = prev = this;
28     }
29     void Unlink() {
30         prev->next = next;
31         next->prev = prev;
32     }
33     void Insert(DbgBlkHeader *lnk) {
34         lnk->prev = this;
35         lnk->next = next;
36         next->prev = lnk;
37         next = lnk;
38     }
39 };
40
41 static const char *DbgFormat(char *b, DbgBlkHeader *p)
42 {
43     sprintf(b, "--memory-breakpoint__ %u ", (dword)~(p->serial ^ (uintp
44     return b;
45 }
46
47 static void DbgHeapPanic(const char *text, DbgBlkHeader *p)
48 {
49     char h[256];
50     char b[100];
51     strcpy(h, text);
52     strcat(h, DbgFormat(b, p));
53     HeapPanic(h, p + 1, (int) (uintptr_t)p->size);
54 }
55
56 static DbgBlkHeader dbg_live = { 0, &dbg_live, &dbg_live, 0 };
57
58 static dword s_allocbreakpoint;
```

3) [oracle_test.upp](#), downloaded 388 times