

\$SPAD/src/lib util.c

The Axiom Team

July 31, 2014

Abstract

Contents

1 License	5
-----------	---

__ * __

```
#include <stdlib.h>
```

The MACOSX platform is broken because no matter what you do it seems to include files from `[[/usr/include/sys]]` ahead of `[[/usr/include]]`. On linux systems these files include themselves which causes an infinite regression of includes that fails. GCC gracefully steps over that problem but the build fails anyway. On MACOSX the `[[/usr/include/sys]]` versions of files are badly broken with respect to the `[[/usr/include]]` versions.

__ * __

```
#if defined(MACOSXplatform)
#include "/usr/include/unistd.h"
#else
#include <unistd.h>
#endif
#include <sys/types.h>
#include <stdio.h>
#include <errno.h>
#include <X11/Xlib.h>
#include <X11/Xutil.h>
#include "view.h"

#include "util.h1"

int
checker(int code, int lineNumber, char *errorStr)
{
    if (code < 0) {
        fprintf(stderr, "Error occurred during %s\n", errorStr);
        fprintf(stderr, "Error code of %d\n", errno);
        fprintf(stderr, "Error in line number %d of process %d\n",
            lineNumber, getpid());
        perror("");
    }
    return (code);
}

char *
getmemWithLine(int nbytes, char *str, int lineNum)
{
    char *p;

    p = (char *) malloc(nbytes);
    if (!p) {
```

```

        fprintf(stderr,
            "getmem: Could not get %d bytes for %s at line %d\n",
            nbytes, str, lineNum);
        exit(99);
    }
    return p;
}

char *
saymemWithLine(char *str, int num, int size, int lineNum)
{
    char *p;
    p = getmemWithLine(num * size, str, lineNum);
    return p;
}

void
myfree(void *p, int size)
{
    free(p);
}

XPoint
getWindowPositionXY(Display *display, Window w)
{
    XPoint position;
    Window rootW, parentW, *childrenWs, tmpW;
    unsigned int nChildren;
    XWindowAttributes windowAttrib;
    int screen, tmp = 1;

    screen = DefaultScreen(display);
    tmpW = w;
    while (tmp) {
        XQueryTree(display, tmpW, &rootW, &parentW, &childrenWs, &nChildren);
        XFree((char *)childrenWs);
        if (parentW == RootWindow(display, screen))
            tmp = 0;
        else
            tmpW = parentW;
    }
    XGetWindowAttributes(display, tmpW, &windowAttrib);
    position.x = (short) windowAttrib.x;
    position.y = (short) windowAttrib.y;
    return (position);
}

XPoint
getWindowSizeXY(Display *display, Window w)
{

```

```

XPoint size;
Window rootW, parentW, *childrenWs, tmpW;
unsigned int nChildren;
XWindowAttributes windowAttrib;
int screen, tmp = 1;

screen = DefaultScreen(display);
tmpW = w;
while (tmp) {
    XQueryTree(display, tmpW, &rootW, &parentW, &childrenWs, &nChildren);
    XFree((char *)childrenWs);
    if (parentW == RootWindow(display, screen))
        tmp = 0;
    else
        tmpW = parentW;
}
XGetWindowAttributes(display, tmpW, &windowAttrib);
size.x = (short) windowAttrib.width;
size.y = (short) windowAttrib.height;
return (size);
}

```

1 License

/*

Copyright (c) 1991-2002, The Numerical ALgorithms Group Ltd.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.
- Neither the name of The Numerical ALgorithms Group Ltd. nor the
names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A

PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

References

- [1] nothing