
Subject: U++ can't handle float to string and back for large numbers

Posted by [cbpporter](#) on Tue, 21 Mar 2017 08:57:45 GMT

[View Forum Message](#) <> [Reply to Message](#)

I'm working on a library that tries to do a lot for things, including float work.

I real Maths work, where you do more than $1.0 + 2.0$, you often encounter large numbers.

I noticed that U++ can't handle the extremes at all, both by debugging the float values and by using CParser extensively. I looked over CParser code extensively and have not yet found the problem. Most if not all the float to string bits form Core have this problem.

Even something as simple as:

```
double d = 1.79769e+302;
printf("%e\n", d); // good
Cout() << d << "\n"; // wrong
```

Is is wrong at it gets worse as worse as you go up to e303 and up to e308.

I am currently investigating this, since U++ problems with "atof" are also my libraries problems by inheritance.

PS: My official release day was planned for Wednesday, but it will probably be moved back to Friday.

Why do such weird problems only happen before release day?

Subject: Re: U++ can't handle float to string and back for large numbers

Posted by [cbpporter](#) on Tue, 21 Mar 2017 09:28:31 GMT

[View Forum Message](#) <> [Reply to Message](#)

Better test case: 1e308.

After further investigations, it looks like CParser is good. So "atof" is pretty much good I think.

DbIStr is to blame, the "ftoa" part. I'm investigating that right now.

Subject: Re: U++ can't handle float to string and back for large numbers

Posted by [cbpporter](#) on Tue, 21 Mar 2017 09:52:46 GMT

[View Forum Message](#) <> [Reply to Message](#)

Still under investigation, but:

```
double normalize(double d, int& exp)
{
    if(!IsNull(d) || d == 0)
    {
        exp = Null;
        return d;
    }
    bool sign = (d < 0);
    if(sign) d = -d;
    exp = minmax<int>(ilog10(d), -308, +308); // 8-byte double!
    d /= ipow10(exp);
    if(d >= 10) { d /= 10; exp++; }
    if(d < 1) { d *= 10; exp--; }
    return sign ? -d : d;
}
```

Looks like fixes the problem. This, and adding special case to suppress the inherent U++ IsNull mechanic. And some custom work to handle infinite better.

Subject: Re: U++ can't handle float to string and back for large numbers
 Posted by [mr_ped](#) on Tue, 21 Mar 2017 20:35:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

I'm not IEEE expert, but why "exp = minmax<int>(ilog10(d), -308, +308);", I mean why decimal exp, when the native is binary, if I understand the IEEE format correctly. The whole code looks weird, all the computations instead of working directly with bits, this will be very likely prone to some rounding/cutoffs artefacts. Also doing it directly from bits with decimal conversions only in the last phase would be probably faster, not just more accurate.

Subject: Re: U++ can't handle float to string and back for large numbers
 Posted by [cbpporter](#) on Wed, 22 Mar 2017 06:27:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

Yeah, I'm no expert either. And probably 308 is not enough to handle subnormal values.

But it is enough for my needs for now. Without this increase, I can't print a single double and neither can't all of U++, including the debugger.

So the debugger fails, DbIStr fails and I need to ftoa the values both in the production code and for debugging.

Subject: Re: U++ can't handle float to string and back for large numbers

Posted by [mirek](#) on Sat, 15 Apr 2017 09:25:02 GMT

[View Forum Message](#) <> [Reply to Message](#)

cbpporter wrote on Tue, 21 March 2017 10:52 Still under investigation, but:

```
double normalize(double d, int& exp)
{
    if(IsNull(d) || d == 0)
    {
        exp = Null;
        return d;
    }
    bool sign = (d < 0);
    if(sign) d = -d;
    exp = minmax<int>(ilog10(d), -308, +308); // 8-byte double!
    d /= ipow10(exp);
    if(d >= 10) { d /= 10; exp++; }
    if(d < 1) { d *= 10; exp--; }
    return sign ? -d : d;
}
```

Looks like fixes the problem. This, and adding special case to suppress the inherent U++ IsNull mechanic. And some custom work to handle infinite better.

Thanks, fix applied. This code is ancient, I guess at that time, Tom has not considered handling 'exteremes' so important.

Subject: Re: U++ can't handle float to string and back for large numbers

Posted by [cbpporter](#) on Thu, 20 Apr 2017 08:44:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

Thank you Mirek!

Now I only need to get rid of <http://ultimatepp.org/redmine/issues/1650> and I can update U++ again .
