Subject: problem with fprintf
Posted by o_wild on Mon, 11 Oct 2010 12:08:10 GMT
View Forum Message <> Reply to Message

The following code is excerpt of an opensource software, but there is error when compiled.

fprintf(ofp,"%s\n",mes1); // mes1 is of type String

The error message is
cannot pass objects of non-trivially-copyable type 'class Upp::String' through '...'

The software can be compiled with both mingw and MSVC as the developer says. Since I have problem using MSVC due to C2001 error(new line in constant), I choose MingGW 4.5.0.

Is their any substitute for 'fprintf'? by the way, i tested 'printf' and found the same problem.

Subject: Re: problem with fprintf
Posted by cbpporter on Mon, 11 Oct 2010 12:26:11 GMT
View Forum Message <> Reply to Message

The problem is that when trying to pass your string as an argument ta a variable argument function the compiler has no means of knowing that you are trying to pass a C string. In such cases you must explicitly specify that you want to convert from String to C string. Try ~mes1, mes1.Begin() or casting to (char*).

Subject: Re: problem with fprintf
Posted by o_wild on Wed, 13 Oct 2010 10:54:05 GMT
View Forum Message <> Reply to Message

Thanks. Both will do.

Subject: Re: problem with fprintf
Posted by mirek on Fri, 15 Oct 2010 12:44:59 GMT
View Forum Message <> Reply to Message

o_wild wrote on Mon, 11 October 2010 08:08The following code is excerpt of an opensource software, but there is error when compiled.

fprintf(ofp,"%s\n",mes1); // mes1 is of type String

Is their any substitute for 'fprintf'? by the way, i tested 'printf' and found the same problem.

Yes, Format. It is typesafe - you would not encounter above kind of problem there.

So in "true" U++ the code would look like this:

```
FileOut ofp;
....
ofp.Put(Format("%d\n", mes1);
```
