Error Checking in FINCAD® Developer Version 9

The latest version of FINCAD Developer provides some enhancements which make it easier then ever before to identify problems with function arguments that cause functions to fail. To download a trial version of FINCAD Developer v9, please visit
http://www.fincad.com/download.asp?id=13900&s=Downloads&n=FINCAD%20Developer

The new scheme associates error codes with input restrictions, when a given function returns an error code, you simply lookup the code on the function reference page to identify what is wrong with the function inputs. The error codes are listed under the restrictions column on the function reference page. The codes are preceded by the #symbol.

aaAccrual_factor

Purpose

Calculate the accrual factor that applies between two dates using the appropriate accrual method (day count basis)

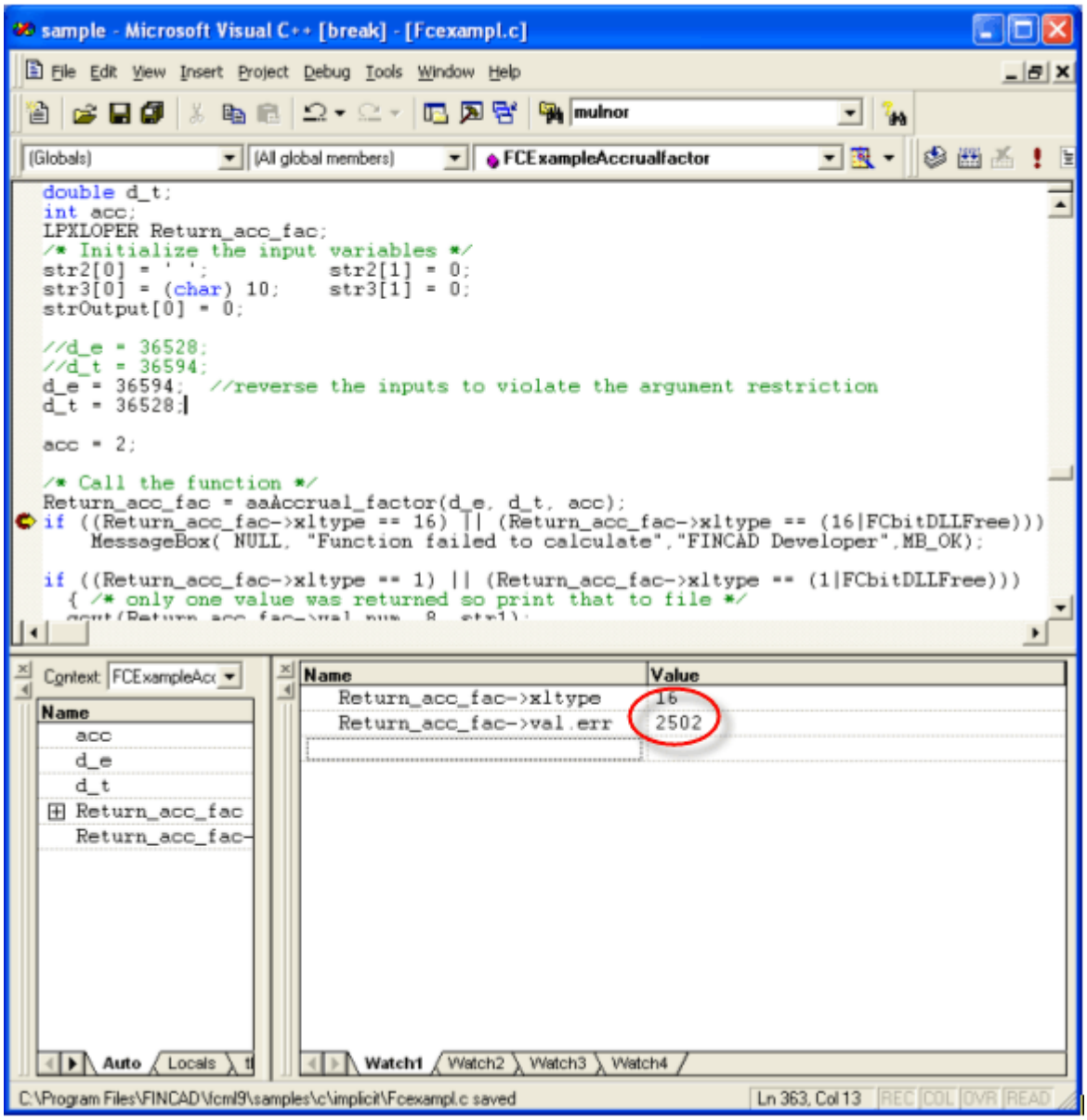
C++ Declaration

LPXLOPER far pascal aaAccrual_factor(double d_e, double d_t, int acc);

Returns	Type	Description
acc_fac	LPXLOPER	accrual factor - day basis/year basis

Argument	Type	Description	Sample	Restriction
d_e	double	effective date	03-Jan-2000	
d_t	double	terminating date	09-Mar-2000	> d_e (#2502)
acc	int	accrual method (sw_331) 1 actual/365 (fixed) 2 actual/360 3 actual/365 (actual) 4 30/360 (ISDA) 5 30E/360 (30/360 ISMA) 6 30E+/360 7 actual/actual (ISMA-99) 8 actual/actual (ISDA) 9 30/360 (old) 10 30E/360 (old) 11 30/360 (SIA) 12 30/360 (BMA) 13 30/360 (German)	2	1..13 (#3793)

Most error codes are either four or five digits. The first one or two digits indicate the position of the function input which has violated an argument restriction. The remaining digits are used to encode information about the specific input validation issue. In the picture above you will notice that the d_t argument has an error code #2502 listed in the restrictions column. The first digit indicates that it is the second argument which has violated a restriction and the remaining digits (502) are an internal code or index which relates the error to the restriction text. In the FINCAD XL product the internal code is processed by the validate inputs feature to return a description of the input restriction that was violated. The example below shows what a developer would see when a function returns an error.



A small number of two digit error codes are used to identify issues that do not relate directly to function arguments. For example, code 25 is used to indicate an expired trial and a code of 15 is used in the c/c++ interface to indicate a generic error. The generic error code may be returned in special cases where no argument restriction was violated but the algorithm fails to converge to a solution given the inputs. The generic code may also be returned in cases where the input validation features have been disabled through a call to the aaErrorHandlingEnable routine or in rare cases when a particular input is not validated due to a software error. A small number of older FINCAD functions have limited input validation for table inputs and may return a generic error code when argument restrictions related to these inputs are violated. There are plans to further enhance input validation in these older functions in version 10. Most of these functions are listed in the function finder in the format aaFunctionName (old).

Additional input validation code in the library is enabled by default in version 9. This means that the feature no longer needs to be enabled by calling the function aaErrorHandlingEnable and passing an input value equal to one. As mentioned previously, the feature can still be disabled by calling the aaErrorHandlingEnable routine and passing a value of zero.

Disclaimer

Your use of the information in this article is at your own risk. The information in this article is provided on an "as is" basis and without any representation, obligation, or warranty from FINCAD of any kind, whether express or implied. We hope that such information will assist you, but it should not be used or relied upon as a substitute for your own independent research.