		ApplicativeTemplatesBase		
OneOperandApplica	ativeTemplatesBase	'	TwoOperandApplicativeTemplatesBase	
		_bz_abs< P_numtype1 >		_bz_Add< P_numtype1, P_numtype2 >
		_bz_abs< double >		_bz_atan2< P_numtype1, P_numtype2 >
		_bz_abs< float >		_bz_atan2< float, float >
		_bz_abs< long >		_bz_atan2< long double, long double >
		_bz_abs< long double >		bz_BitwiseAnd< P_numtype1, P_numtype2 >
		_bz_acos< P_numtype1 >		_bz_BitwiseOr< P_numtype1, P_numtype2 >
		_bz_acos< float >		bz_BitwiseXOR< P_numtype1, P_numtype2 >
		_bz_acos< long double > _bz_asin< P_numtype1 >		bz_Divide< P_numtype1, P_numtype2 >bz_Equal< P_numtype1, P_numtype2 >
		_bz_asin< float >		bz_fmod< P_numtype1, P_numtype2 >
		_bz_asin< long double >		bz_Greater< P_numtype1, P_numtype2 >
		_bz_atan< P_numtype1 >		_bz_GreaterOrEqual< P_numtype1, P_numtype2 >
		_bz_atan< float >		bz_Less< P_numtype1, P_numtype2 >
		_bz_atan< long double >		_bz_LessOrEqual< P_numtype1, P_numtype2 >
		_bz_BitwiseNot< P_numtype >		_bz_LogicalAnd< P_numtype1, P_numtype2 >
		_bz_Cast< P_numtype_in, P_numtype_out >		bz_LogicalOr< P_numtype1, P_numtype2 >
		_bz_ceil< P_numtype1 >		_bz_Max< P_numtype1, P_numtype2 >
		_bz_ceil< float >		_bz_Min< P_numtype1, P_numtype2 >
		_bz_ceil< long double >		_bz_Mod< P_numtype1, P_numtype2 >
		_bz_cexp< P_numtype1 >		_bz_Multiply< P_numtype1, P_numtype2 >
		_bz_cos< P_numtype1 >		bz_NotEqual< P_numtype1, P_numtype2 >
		_bz_cos< float >		_bz_pow< P_numtype1, P_numtype2 >
		_bz_cos< long double >		_bz_pow< float, float >
		_bz_cosh< P_numtype1 > _bz_cosh< float >		_bz_pow< long double, long double > _bz_ShiftLeft< P_numtype1, P_numtype2 >
		_bz_cosh< long double >		bz_ShiftRight< P_numtype1, P_numtype2 >
		_bz_csqrt< P_numtype1 >		bz_Subtract< P_numtype1, P_numtype2 >
		_bz_exp< P_numtype1 >		
		_bz_exp< float >		
		_bz_exp< long double >		
		_bz_floor< P_numtype l >		
		_bz_floor< float >		
		_bz_floor< long double >		
		_bz_log< P_numtype1 >		
		_bz_log10< P_numtype1 >		
		_bz_log10< float>		
		_bz_log10< long double >		
		_bz_log< float > _bz_log< long double >		
		bz_LogicalNot< P_numtype >		
		_bz_negate< P_numtype >		
		_bz_pow2< P_numtype1 >		
		_bz_pow3< P_numtype1 >		
		_bz_pow4< P_numtype1 >		
		_bz_pow5< P_numtype1 >		
		_bz_pow6< P_numtype1 >		
		_bz_pow7< P_numtype1 >		
		_bz_pow8< P_numtype1 >		
		_bz_sin< P_numtype1 >		
		_bz_sin< float >		
		_bz_sinh <p_numtype1></p_numtype1>		
		_bz_sinh< float >		
		_bz_sinh< long double >		
		_bz_sqr< P_numtype >		
		_bz_sqrt< P_numtype1 >		
		_bz_sqrt< float >		
		_bz_sqrt< long double >		
		_bz_tan< P_numtype1 >		
		_bz_tan< float >		
		_bz_tan< long double >		
		_bz_tanh< P_numtype1 >		
		_bz_tanh< float >		
l		_bz_tanh< long double >		