Common 64-bit Function Calling Conventions

		Windows		System V ABI	
Parameters		Integer (including Pointers)	Floating Point	Integer (including Pointers)	Floating Point
	Left-most	rcx	xmm0	rdi	xmm0
	…	rdx	xmm1	rsi	xmm1
	…	r8	xmm2	rdx	xmm2
	…	r9	xmm3	rcx	xmm3
	…	*On stack.*		r8	xmm4
	…			r9	xmm5
	…			*On stack*.	xmm6
	…				xmm7
	… (and onward)				*On stack.*
Return Value		Integer (including Pointers)	Floating Point	Integer (including Pointers)	Floating Point
		rax	xmm0	rax	xmm0, xmm1
Register Preservation:		Integer (including Pointers)	Floating Point	Integer (including Pointers)	
	Callee-Saved	rbx, rbp, rdi, rsi, rsp, r12, r13, r14, r15	xmm6-xmm15	rbp, rbx, r12, r13, r14, r15	

Notes:

1. Linux, the BSDs, Solaris and (mostly) macOS (when it was designed for x86-64 chips) all adhere to the SystemV ABI.
2. "Callee-saved" means that it is the responsibility of the function being invoked to make sure that the designated registers' values are returned to their original state before control returns to the calling function. From the perspective of the function caller, therefore, the values in callee-saved registers are stable through function calls.

3. Sources: System V ABI: https://www.uclibc.org/docs/psABI-x86_64.pdf; Windows: https://learn.microsoft.com/en-us/cpp/build/x64-calling-convention?view=msvc-170