

GCC	LLVM	JIT	Meaning
610: movzbl 0x8(%eax,%edx,1),%ebp	0001f644: movl -0x14(%ebp), %esi	000018b5: movzbl %ds:(%edx,%eax,1), %edi	move the string1 character (at index i) to a register (either %ebp, %esi or %edi)
	0001f647: movzbl 0x8(%edx,%esi), %esi		
615: movzbl 0x8(%eax,%ebx,1),%ecx	0001f64c: movl -0x18(%ebp), %edi	000018b9: movzbl %ds:(%esi,%eax,1), %ecx	move the string2 character (at index i) to a register (either %ecx, %edi or %ecx)
	0001f64f: movzbl 0x8(%edx,%edi), %edi		
61a: cmp %ecx,%ebp	0001f654: subl %edi, %esi	000018bd: subl %ecx, %edi	compare these 2 characters. If they are different, jump after the loop
61c: jne 6e8	0001f656: jne 0x1f660	000018bf: jnz 0x00001900	
622: add \$0x1,%eax	0001f658: incl %edx	000018c1: addl \$0x00000001, %eax	increment the index i
625: cmp %eax,%edi	0001f659: cmpl %ebx, %edx	000018c4: cmpl %eax, %ebx	check if i is not off bounds. If it is still in bounds, jump at the start of the loop
627: jne 610	0001f65b: jl 0x1f644	000018c6: jnz 0x000018b5	