moved from (for output values). This is what defines whether the value is placed in a register or a memory location.

The constraint is a single-character code. The constraint codes are shown in the following table.

Constraint	Description
a	Use the %eax, %ax, or %al registers.
b	Use the %ebx, %bx, or %bl registers.
c	Use the %ecx, %cx, or %cl registers.
d	Use the %edx, %dx, or $dl registers.
S	Use the %esi or %si registers.
D	Use the %edi or %di registers.
r	Use any available general-purpose register.
q	Use either the %eax, %ebx, %ecx, or %edx register.
A	Use the %eax and the %edx registers for a 64-bit value.
f	Use a floating-point register.
t	Use the first (top) floating-point register.
u	Use the second floating-point register.
m	Use the variable's memory location.
o	Use an offset memory location.
V	Use only a direct memory location.
i	Use an immediate integer value.
n	Use an immediate integer value with a known value.
g	Use any register or memory location available.

In addition to these constraints, output values include a constraint modifier, which indicates how the output value is handled by the compiler. The output modifiers that can be used are shown in the following table.

Output Modifier	Description
+	The operand can be both read from and written to.
=	The operand can only be written to.
%	The operand can be switched with the next operand if necessary.
&	The operand can be deleted and reused before the inline functions complete.