```
 1.  vmovsd (%r10),%xmm0                 # Load 1 element of C into %xmm0
 2.  mov    %rsi,%rcx                    # register %rcx = %rsi
 3.  xor    %eax,%eax                    # register %eax = 0
 4.  vmovsd (%rcx),%xmm1                 # Load 1 element of B into %xmm1
 5.  add    %r9,%rcx                     # register %rcx = %rcx + %r9
 6.  vmulsd (%r8,%rax,8),%xmm1,%xmm1     # Multiply %xmm1, element of A
 7.  add    $0x1,%rax                    # register %rax = %rax + 1
 8.  cmp    %eax,%edi                    # compare %eax to %edi
 9.  vaddsd %xmm1,%xmm0,%xmm0            # Add %xmm1, %xmm0
10.  jg     30 <dgemm+0x30>              # jump if %eax > %edi
11.  add    $0x1,%r11                    # register %r11 = %r11 + 1
12.  vmovsd %xmm0,(%r10)                 # Store %xmm0 into C element
```

FIGURE 2.44 The x86 assembly language for the body of the nested loops generated by compiling the unoptimized C code in Figure 2.43 using gcc with −O3 optimization flags.