1.1.3

1. How do assemblers and linkers work together?

Answer: An assembler converts source-code programs from assembly language into machine language, which is referred to as object-code. A linker combines individual files created by an assembler into a single executable program.

1. What is meant by a one-to-many relationship when comparing a high-level language to machine language?

Answer: In a one-to-many relationship, a single statement expands into many assembly language or machine instructions.

8. Do you suppose type checking on pointer variables is stronger (stricter) in assembly language, or in C and C++?

Answer: C++ does not allow a pointer of one type to be assigned to a pointer of another type. On the other hand, Assembly language has no restrictions regarding pointers.

12. Challenge: Translate the following C++ expression to assembly language, using the example presented earlier in this chapter as a guide: X (Y \* 4) 3.

Answer:

mov eax, Y  
mov ebx, 4  
imul, ebx  
add eax, 3  
mov X, eax

1.2.1

2. Why do you suppose translated programs often execute more quickly than interpreted ones?

Answer: Translated programs often execute more quickly than interpreted ones. This is because the translated programs are now converted into a language that can be understood by the machine, and executed with no hassle. On the other hand, interpreted programs are translated and executed simultaneously, causing them to be executed less quickly.

3. (True/False): When an interpreted program written in language L1 runs, each of its instructions is decoded and executed by a program written in language L0.

Answer: True

10. Statements at the assembly language level of a virtual machine are translated into statements at which other level?

Answer: 2nd level

1.3.9

2. What is the decimal representation of each of the following unsigned binary integers?

a. 11111000

b. 11001010

c. 11110000

Answer:

a. 248

b. 202

c. 240

3. What is the sum of each pair of binary integers?

a. 00001111 + 00000010

b. 11010101 + 01101011

c. 00001111 + 00001111

Answer:

a. 00010001

b. 101000000

c. 00011110

5. What is the minimum number of binary bits needed to represent each of the following unsigned decimal integers?

a. 65

b. 409

c. 16385