

Exercise 01

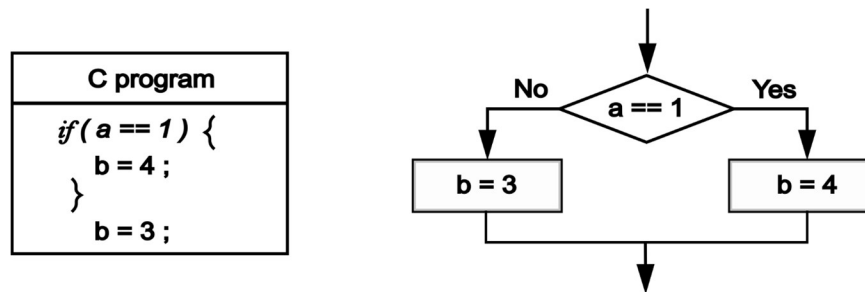
Write code that reads from variable N, adds 10 to it, and stores the result in variable M. Both variables are 32-bit.

Exercise 02

Write a program that reads a 32-bit integer from memory location 0x20000000, performs a bitwise AND operation with the value 0xFFFF0000, and stores the result back into the same memory location.

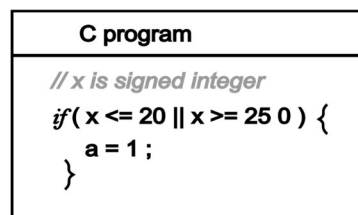
Exercise 03

Write the assembler code corresponding to the following C code.



Exercise 04

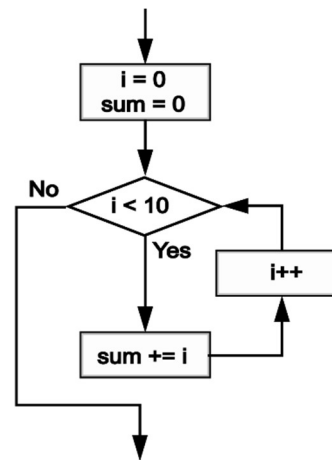
Write the assembler code corresponding to the following C code.



Exercise 05

Write the assembler code corresponding to the following C code.

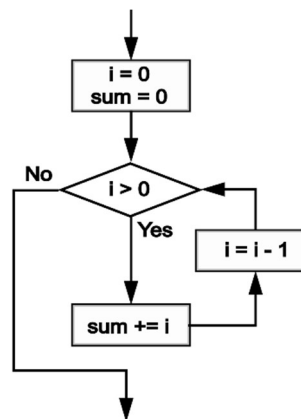
C program
<pre> int = i; int = sum; if (i = 1; i < 10; i++) { sum += i; } </pre>



Exercise 06

Write the assembler code corresponding to the following C code.

C program
<pre> int = i; int = sum; while (i > 0) { sum += i; i--; } </pre>



Exercise 07

Write a program that initializes an array of 5 integers with values 10, 20, 30, 40, and 50, starting from memory address 0x20002000. Then, multiply each element of the array by 2 and store the results back into the same memory locations.

Exercise 08

Initialize an array of 10 integers in memory starting from address 0x20001000 with values 1, 2, 3, ..., 10. Use a loop construct to perform this initialization.