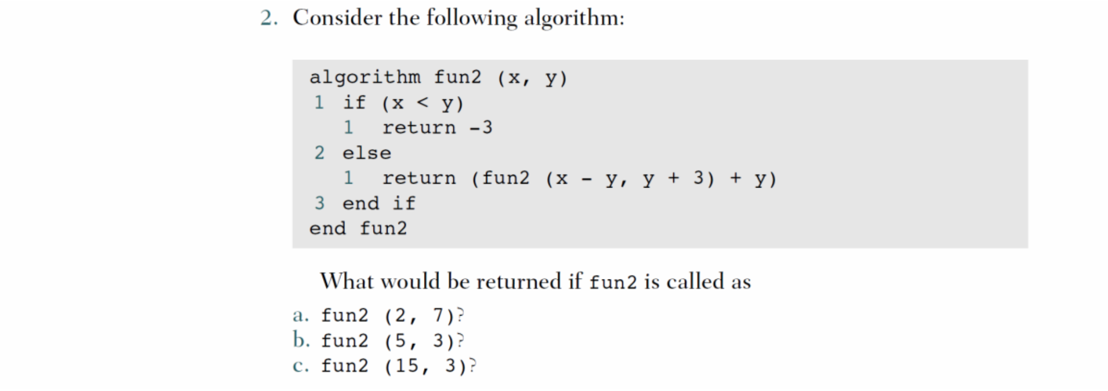
HW2

(deadline 2017/10/13)

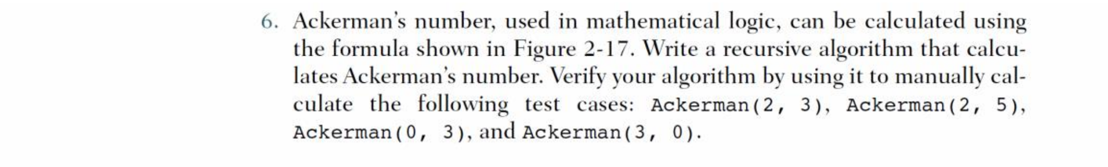


2. Consider the following algorithm:

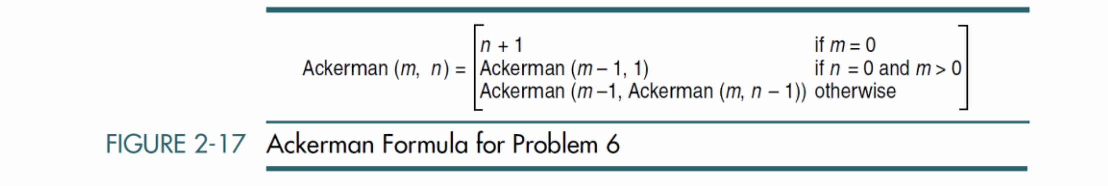
```
algorithm fun2 (x, y)
1 if (x < y)
   1  return -3
2 else
   1  return (fun2 (x - y, y + 3) + y)
3 end if
end fun2
```

What would be returned if `fun2` is called as

a. `fun2 (2, 7)`?
b. `fun2 (5, 3)`?
c. `fun2 (15, 3)`?



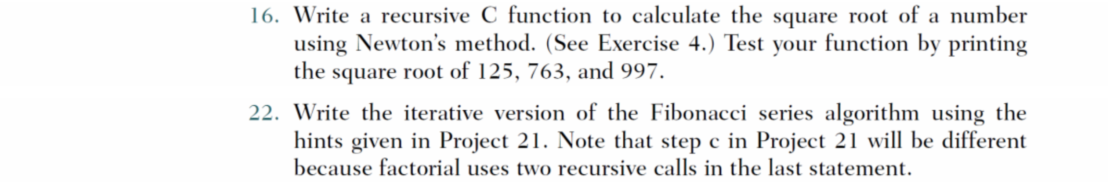
6. Ackerman's number, used in mathematical logic, can be calculated using the formula shown in Figure 2-17. Write a recursive algorithm that calculates Ackerman's number. Verify your algorithm by using it to manually calculate the following test cases: `Ackerman(2, 3)`, `Ackerman(2, 5)`, `Ackerman(0, 3)`, and `Ackerman(3, 0)`.



$$\text{Ackerman}(m, n) = \begin{bmatrix} n + 1 & \text{if } m = 0 \\ \text{Ackerman}(m - 1, 1) & \text{if } n = 0 \text{ and } m > 0 \\ \text{Ackerman}(m - 1, \text{Ackerman}(m, n - 1)) & \text{otherwise} \end{bmatrix}$$

FIGURE 2-17 Ackerman Formula for Problem 6

程式題:



16. Write a recursive C function to calculate the square root of a number using Newton's method. (See Exercise 4.) Test your function by printing the square root of 125, 763, and 997.

22. Write the iterative version of the Fibonacci series algorithm using the hints given in Project 21. Note that step c in Project 21 will be different because factorial uses two recursive calls in the last statement.