High-level Language Programmer’s View

main:
maxNum = 3
maxPower = 4

CalculatePowers(maxNum, maxPower)
(*)

... end main

CalculatePowers(In: integer numLimit, integer powerLimit)

integer result, num, pow

for num := 1 to numLimit do
  for pow := 1 to powerLimit do
    result = Power(num, pow, result)
    (**)
    print num “ raised to “ pow “ power is “ result
  end for pow
end for num

end CalculatePowers

integer Power(In: integer n, integer e)

integer result

if e = 0 then
  result = 1
else if e = 1 then
  result = n
else
  result = Power(n, e - 1) * n
  (***)
end if

return result

end Power

1) Trace the next execution of the recursive function Power by showing the run-time stack.

2) What is the most number of call frames on the stack at any one time for the whole program?