High-level Language Programmer’s View

main:
maxNum = 3
maxPower = 4

CalculatePowers(maxNum, maxPower)
(*)
...
end main

CalculatePowers(In: integer numLimit, integer powerLimit)
integer num, pow, result

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

end CalculatePowers

Power( In: integer n, integer e, Out: result)

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

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?
Assembly-language Programmer’s View

3) Trace the hypothetical assembly language program and indicate the resulting value of the registers Reg1, Reg2, Reg3, and Reg4.

```
.data
X: .WORD 2 ; variable X initialized at assembly time to 2
Y: .WORD 3 ; variable Y initialized at assembly time to 3
Z: .WORD 0 ; variable Z initialized at assembly time to 0

.prog
Begin:
    LOAD Reg1, X ; loads X’s value into register Reg1
    LOAD Reg2, Y
    ZERO Reg3 ; sets Reg3’s value to 0
    MOVE Reg4, Reg2 ; Reg4 := Reg2
Loop:
    ADD Reg3, Reg3, Reg1 ; Reg3 := Reg3 + Reg1
    SUB_IMMEDIATE Reg4, Reg4, #1 ; Reg4 := Reg4 - 1
    BRANCH_GREATERTHANZERO Reg4, Loop ; if Reg4 > 0 then goto Loop label
    STORE Reg3, Z ; store Reg3’s value into variable Z
End:
```

<table>
<thead>
<tr>
<th>Resulting register values</th>
<th>Reg1</th>
<th>Reg2</th>
<th>Reg3</th>
<th>Reg4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

a) What is the resulting value in Z?

b) What calculation does this code perform?

4) During the execution of the above assembly language code: (Assuming no cache)
a) How many memory reads were performed? (state any assumptions)
   \[# \text{data reads} = \# \text{instruction reads (assume one read per instruction fetch)} = \]

b) How many memory writes were performed? (state any assumptions)

5) List (in decreasing order of importance) why somebody would write assembly language code.
   (top reason) a) 
   b) 
   c)