Task 1:

1. 2logn
2. 4n
3. 3n+100logn
4. 4nlogn + 2n
5. nlogn
6. n2 + 10n
7. 210
8. 2n
9. n3

Task 2:

Cách 1: function power\_of\_two(n)

result = 1

for i = 1 to n

result = result \* 2

return result

end function

cách 2: function power\_of\_two(n)

return 1 << n

end function

Task 3:

function enqueue(queue[], item)

if queue is not full

add item to the end of the queue

else

print "Queue is full"

end if

end function

Task 4:

function enqueue(queue, item)

create a new node with value item

if queue is empty

set both head and tail of the queue to the new node

else

set the next pointer of the tail to the new node

update the tail to the new node

end if

end function

Task 5:

function isFull(stack[])

if top of stack is equal to the maximum index (indicating a full stack)

return true

else

return false

end if

end function

Task 6:

function push(stack, item)

create a new node with value item

set the next pointer of the new node to the top of the stack

update the top of the stack to the new node

end function