

Define Stack A

Define Stack B

counter = 0

i = 0

while (i < n):

height = Input()

if (i == 0):

counter += 1

Stack A.push(height)

else:

CanSee = True

while (!Stack A.is Empty())

temp = Stack A.pop()

if (height < temp):

CanSee = False

Stack B.push(temp)

while (!Stack B.is Empty())

Stack A.push(Stack B.pop())

Stack A.push(height)

i += 1

Print (Counter) ← جواب

Define Stack A

Define Stack B

Function Enqueue(person):

StackA.push(person)

Function Dequeue:

if (StackA.isEmpty()):

print("Queue is empty :)")

else:

while (!StackA.isEmpty()):

StackB.push(StackA.pop())

result = StackB.pop()

while (!StackB.isEmpty()):

StackA.push(StackB.pop())

return result

3) برای یک Linked List که head دارد، داریم:

$p = \text{head}$
 $q = \text{head}$

while (True):

$p = p.\text{next}$

if ($q.\text{next} \neq \text{null}$):

$q = q.\text{next}.\text{next}$

else:

return False

if ($p == \text{null}$ or $q == \text{null}$):

return False

if ($p == q$):

return True

اگر در Linked List یک لایه
 اضافه شده باشد، الگوریتم ما
 برمیگردد.

4) این 3 تا پوینتر به این صورت است: current ptr c، previous ptr p، next ptr n

انتخاب کنیم و داریم:

$c = \text{head}$

$p = \text{NULL}$

$n = \text{NULL}$

while ($c \neq \text{NULL}$):

$n = c.\text{next}$

$c.\text{next} = p$

$p = c$

$c = n$

head = p