Lab W1D6

Q1,

a, Algorithm partition(S)

Input sequence S with un ordered blue and Red wooden blocks with n size

Output S in ordered order of Red and Blue

if(|S|=0 or |S|=1) then return S

R, B ← empty sequences

while !S.isEmpty()

y ← S.removeFirst()

if y = RED

R.insertLast(y)

else

B.insertLast(y)

return R U B

b,Algorithm partition(S)

Input sequence S with un ordered blue and Red wooden blocks with n size

Output S in ordered order of Red and Blue

if(|S|=0 or |S|=1) then return S

R, B ,G← empty sequences

while !S.isEmpty()

y ← S.removeFirst()

if y = RED

R.insertLast(y)

else if y=GREEN

G.insertLast(y)

else

B.insertLast(y)

return R U G U B

C, Algorithm partition(S)

Input sequence S with un ordered blue and Red wooden blocks with n size

Output S in ordered order of Red and Blue

if(|S|=0 or |S|=1) then return S

R, B ,Y,G← empty sequences

while !S.isEmpty()

y ← S.removeFirst()

if y = RED

R.insertLast(y)

else if y=GREEN

G.insertLast(y)

else if y=YELLOW

Y.insertLast(y)

else

B.insertLast(y)

return R U G U Y U B

Time complexity for all 3 algorithms is O(n).

Q2,

A, 1,2,3,4,5,6,7,8,9

Since j is less than I array is sorted.

B