Q2,

We can use mergeSort so that we can have exactly 5 comparisons for 4 elements.

Bubble, selection and insertion sort take more than 5 comparisons for 4 elements.

Algorithm mergeSort(S)

Input sequence S with 4

Output sequence S sorted

if S.size() > 1 then

(S1 , S2 ) ← partition(S, n/2)

mergeSort(S1 )

mergeSort(S2 )

S ← merge(S1 , S2 )

return S

Algorithm merge(A, B)

Input sorted sequences A and B with n/2 integers each

Output sorted sequence S of A ∪ B

S ← empty sequence

while ¬A.isEmpty() ∧ ¬B.isEmpty() do

if A.first() <= B.first() then

S.insertLast(A.remove(A.first()))

else

S.insertLast(B.remove(B.first()))

while ¬A.isEmpty() do

S.insertLast(A.remove(A.first()))

while ¬B.isEmpty() do

S.insertLast(B.remove(B.first()))

return S