**Bubble Sort:**

procedura bubble\_sort (**array** //array to order)

var change<-true

while(change is true) do

change<-false //iterate while index variable has not reached array limit

for (var index<-0 a dimensione\_array(Array) – 2, index+1) do

//iterate until index until element before pultimate index hasn’t been reached

If(valore array in posizione index>valore array in posizione index+1) then

swap(**valore array in posizione index, valore array in posizione index+1**)

change<-true

**Merge Sort:**

procedure merge\_sort (**array** //array to order, **left\_index** //first index in array, **right\_index** //last index in array)

if(right\_index>1)

var middle\_index<-(left\_index+right\_index)/2

// middle index is used for array halving

merge\_sort(**array, left\_index, middle\_index**)

merge\_sort(**array, middle\_index+1, right\_index**)

//divide and iterate Array halves

merge(**array, left\_index, middle\_index, right\_index**)

//procedure that merges sorted arrays value (swaps elements if needed)