**Selection Sort**

**Smallest Sort(Arr,n,i,pos)**

1. Set small=Arr[i]
2. Set Pos=i
3. Repeat for j=i+1 to n-1

If small=Arr[j]

Set Pos=j

1. Return Pos

**Selection Sort(Arr,n)**

1. Repeat steps 2 and 3 for i=1 to n-1
2. Call Smallest(Arr,n,i,Pos)
3. Swap Arr[i] with Arr[Pos]

**Merge Sort**

**MERGE (ARR, BEG, MID, END)**

1. [INITIALIZE] SET I = BEG, J = MID + 1, INDEX =0
2. Repeat while (I <= MID) AND (J<=END)

IF ARR[I] < ARR[J]

SET TEMP[INDEX] = ARR[I]

SET I = I + 1 ELSE

SET TEMP[INDEX] = ARR[J]

SET J = J + 1

[END OF IF]

SET INDEX = INDEX + 1

[END OF LOOP]

1. [Copy the remaining elements of right sub-array, if any]

IF I > MID Repeat while J <= END

SET TEMP[INDEX] = ARR[J]

SET INDEX = INDEX + 1, SET J = J + 1

[END OF LOOP]

[Copy the remaining elements of left sub-array, if any]

ELSE Repeat while I <= MID SET TEMP[INDEX] = ARR[I]

SET INDEX = INDEX + 1, SET I = I + 1

[END OF LOOP]

[END OF IF]

1. [Copy the contents of TEMP back to ARR]

SET K= 0

1. Repeat while K < INDEX

SET ARR[K] = TEMP[K]

SET K = K + 1

[END OF LOOP]

1. END

**MERGE\_SORT(ARR, BEG, END)**

1. IF BEG < END SET MID = (BEG + END)/2

CALL MERGE\_SORT (ARR, BEG, MID)

CALL MERGE\_SORT (ARR, MID + 1, END)

MERGE (ARR, BEG, MID, END)

[END OF IF]

1. END