MERGE SORT

```
Algorithm Merge(A, low, mid, high)
{
      h := low;
      i := low;
      j := mid + 1;
      while((h <= mid) and (j <= high)) do
      {
            if(A[h] <= A[j]) then
            {
                  B[i] := A[h];
                  h := h + 1;
            }
            else
            {
                  B[i] := A[j];
                  j := j + 1;
            }
            i := i + 1;
      }
      if(h > mid) then
      {
            for k := j to high do
            {
                  B[i] := A[k];
                  i := i + 1;
            }
      }
      else
      {
            for k := h to mid do
            {
                  B[i] := A[k];
                  i := i + 1;
            }
      }
      for k := low to high do
            A[k] := B[k];
}

Algorithm MergeSort(A, low, high)
{
      if(low < high) then
      {
            mid := (low + high)/2;
            MergeSort(A, low, mid);
            MergeSort(A, mid+1, high);
            Merge(A, low, mid, high);
      }
}
```