2.3 Merge Sort

Write a program `merge_sort.cpp` OR `merge_sort.py` that implements and demonstrates the merge and merge sort pseudocode on pages 31 and 34.

Specifications: Code and utilize the following functions:

Listing 1: print vector for .cpp only

```
/*
* print_vector(v)for merge_sort.cpp ONLY
*  takes integer vector v as a const reference parameter
*  Prints the contents of vector v.
*  v is not modified
*/
```

Listing 2: merge

```
/*
* merge(A, p, q,  r)
*   takes integer vector A as a reference parameter
*   subarray L = A[p..q] and R = A[q+1..r] are sorted
*   merge L = A[p..q] and R = A[q+1..r] into sorted subarray A[p..r]
*   A IS modified
*/
```

Listing 3: merge sort

```
/*
* merge_sort(A, p, r)
*   takes integer vector A as a reference parameter
*   subarray A[p..r] is the vector to sort
*   A IS modified
*/
```

Listing 4: main

```
/*
* main()
*   Demonstrate merge_sort(A, 0, n-1)
*/
```

Listing 5: merge sort example run

```
The vector to be sorted: {5,2,4,6,1,3,4,2,22,1}

The vector after merge sort: {1,1,2,2,3,4,4,5,6,22}
```