```
Enter number of elements
8
Enter 8 elements
1
4
6
3
2
8
7
6
Sorted list in ascending order:
1
2
3
4
6
6
7
8
```

Algorithm

Step 1: Start

Step 2: Declare void bubble sort (long[], long)

Step 3: Declare long array [100], n, c, d, swap;

Step 4: Read n value

Step 5: Read and display array [c] values using for loop.

Step 6: Void bubble sort (long list [], long n)

Step 7: Declare long c, d, t

   7.1 for(c=0; c<(n-1); c++)

      for(d=0; d< n-c-1; d++)

        if (list [d] > list [d+1])

          t = list [d]

          list [d] = list [d+1]

          list [d+1] = t

Step 8: Stop.

```
( Start )
        │
        ▼
┌──────────────────────────────────────────┐
│ Declare void bubble _sort (Long [], long │      Calling
└──────────────────────────────────────────┘       function
        │
        ▼
┌──────────────────────────────────────────┐
│ Declare Long array [100], n, c, d, swap  │
└──────────────────────────────────────────┘
        │
        ▼
    / Read n Value /
        │
        ▼
/ Read and display array [c] using for loop /
        │
        ▼
┌──────────────────────────────────────────┐
│ Void bubble sort (Long list [], Long n)  │      Called
└──────────────────────────────────────────┘       function
        │
        ▼
┌──────────────────────────┐
│ Declare Long c, d, t     │
└──────────────────────────┘
        │
        ▼
< for ( c = 0; c < (n-1); c++ >
        │ T
        ▼
< for ( d = 0; d < n-c-1; d++ >
        │ T
        ▼
  < if (list [d] > list [d+1] >
        │ T
        ▼
┌──────────────────────────┐
│   t = list [d]           │
│                          │
│  list [d] = list [d+1]   │
│                          │
│  list [d+1] = t          │
└──────────────────────────┘
        │ t
        ▼
    ( Stop )
```