```
695  //Merge Sort
696  #include<stdio.h>
697  #include<stdlib.h>
698  #include<time.h>
699  void merge(int a[],int l,int mid,int r)
700  {
701      int c[10000],i,j,k;
702      i=k=l;
703      j=mid+1;
704      while(i<=mid&&j<=r){
705          if(a[i]<a[j]){
706              c[k]=a[i];
707              ++k;
708              ++i;
709          }
710          else{
711              c[k]=a[j];
712              ++k;
713              ++j;
714          }
715      }
716      if(i>mid){
717          while(j<=r){
718              c[k]=a[j];
719              ++k;
720              ++j;
721          }
722      }
723      if(j>r){
724          while(i<=mid){
725              c[k]=a[i];
726              ++k;
727              ++i;
```

```
728             }
729         }
730         for(i=l;i<=r;i++)
731             a[i]=c[i];
732     }
733     void mergeSort(int a[],int l,int r)
734     {
735         int mid;
736         if(l<r){
737             mid=(l+r)/2;
738             mergeSort(a,l,mid);
739             mergeSort(a,mid+1,r);
740             merge(a,l,mid,r);
741         }
742     }
743     int main(){
744         int a[10000],n,i;
745         printf("Enter the number of elements : ");
746         scanf("%d",&n);
747         printf("\nEnter array elements :\n");
748         for(i=0;i<n;i++)
749             a[i]=rand();
750         clock_t begin=clock();
751         mergeSort(a,0,n-1);
752         clock_t end=clock();
753         double m_time=0.0;
754         m_time+=(double)(end-begin)/CLOCKS_PER_SEC;
755         printf("\nSorted array : ");
756         for(i=0;i<n;i++)
757             printf("%d ",a[i]);
758         printf("\n\nTime taken for %d numbers is %f",n,m_time);
759         return 0;
760     }
```

```
Enter the number of elements : 5

Enter array elements :

Sorted array : 846930886 1681692777 1714636915 1804289383 1957747793

Time taken for 5 numbers is 0.000004

...Program finished with exit code 0
Press ENTER to exit console.
```

	Array Values		
Array Size	Random	Ascending	Descending
100	0.000022	0.000014	0.000014
200	0.000037	0.000017	0.000017
300	0.000057	0.000027	0.000027
400	0.000084	0.000038	0.000035
500	0.000092	0.000056	0.000046
1000	0.000236	0.000105	0.000104
5000	0.001079	0.000651	0.000596
10000	0.001996	0.001402	0.001215

Merge Sort