Program 5

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int len,a[1000],count=0;

void heapify(int i)
{

      int heap=0,j,v,k;

        k=i;
        v=a[k];  // v is the key value

      while (!heap && 2*k<=len)
      {

            j=2*k;
            if (j<len)
            {        count ++;
                  if (a[j]<a[j+1])
                        j=j+1;
            }
                count ++;
            if (v>=a[j])
                  heap=1; // heapification ends
            else
            {
                  a[k]=a[j];  // shift  a[j] up  ie to parents position
                  k=j;
            }
      }
      a[k]=v;
}

void heap()
{
      int i,t;
```

```
        // initial heapification
      for (i=len/2;   i>0;  i--)
      {
            heapify(i);
      }

        // sorting logic
      for (i=len; i>1 ;i--)
      {
            count++;
            t=a[len];
            a[len]=a[1];
            a[1]=t;
            len--;
            heapify(1);
      }
}

int main()
{
      int i,j,n,c1,c2,c3;
      printf("Enter no. of elements:");
      scanf("%d",&n);
      len=n;
      printf("\nEnter %d elements:\n",n);
      for (i=1;i<=len;i++)
            scanf("%d",&a[i]);
      heap();
      printf("\nSorted array is \n\n");
      for (i=1;i<=n;i++)
            printf("%d ",a[i]);
      printf("\n\n no of counts:%d\n",count);
      printf("\n\nsize\tasc\tdesc\trandom\n");
      for (i=16;i<520;i*=2)
      {
            count=0;
            len=i;
            for (j=1;j<=i;j++)
                  a[j]=j;
            heap();
            c1=count;
            count=0;
            len=i;
            for (j=1;j<=i;j++)
                  a[j]=i-j;
            heap();
            c2=count;
            count=0;len=i;
            for (j=1;j<=i;j++)
                  a[j]=rand()%i;
            heap();
            c3=count;
            printf("\n%d\t%d\t%d\t%d",i,c1,c2,c3);
      }
      return 0;
}
```