/*C Program to implement Heap Construction [*bottom-up approach*]

Input : 1. Size of the Array[n]

2. Array elements

Output : A heap with the array elements

***/**

```
#include<stdio.h>
#include<math.h>
main()
{
   int n,a[50],i,j,k,v,heap;
   printf("\n  How many numbers you want to have in the heap?\n ");
   scanf("%d",&n);
   printf("\n  Enter the list of numbers you want to add.\n ");
   for(i=0;i<=n;i++)
   {
      if(i==0)      //Leave index-zero unused
      {
         a[i] = 999;
      }
      else
      {
      scanf("%d",&a[i]);
      }
   }

   for(i=floor(n/2);i>=1;i--)
   {
      k=i;
      v=a[k];
```

```
        heap = 0;
        while(!heap && 2*k <=n)
        {
           j = 2*k;
           if(j<n)    // When there are two children
           {
              if(a[j] < a[j+1])
              j = j+1;
           }
           if(v>=a[j])  //When parental dominance already holds
           {
              heap = 1;
           }
           else        //Swap the parental node with largest child-node
           {
              a[k] = a[j];
              k = j;
           }

        } //end of while loop
        a[k]=v;
     }//end of for loop


   printf("\n\n  Numbers in the heap in the top-down and left-right fashion
are\n\n  ");
   for(k=1;k<=n;k++)
   {
        printf("%d ",a[k]);
   }
   printf("\n\n\n  ");
}
```

Sample Input and Output:

```
How many numbers you want to have in the heap?
8

Enter the list of numbers you want to add.
25 57 48 37 12 92 86 33


Numbers in the heap in the top-down and left-right fashion are

92 57 86 37 12 48 25 33


Press any key to continue...
```