Name: Tushar Basak
ID:1911858042
Sec:02,CSE373

Procedures:

MinHeapify(A, i)

1. l ← left(i)

2. r ← right(i)

3. **if** l ≤ heap-size[A] **and** A[l] < A[i]

4. **then** smallest ← l

5. **else** smallest ← i

6. **if** r ≤ heap-size[A] **and** A[r] <A[smallest]

7. **then** smallest ← r

8. **if** smallest≠ i

9. **then** exchange A[i] ↔ A[smallest]

10. MinHeapify(A, smallest)

BuildMinHeap(A)

1. heap-size[A] ← length[A]

2. **for** i ← ⌊length[A]/2⌋ **downto** 1

3. **do** MinHeapify(A, i)

HeapSort(A)

1. Build-Min-Heap(A)

2. **for** i ← length[A] **downto** 2

3. **do** exchange A[1] ↔ A[i]

4. heap-size[A] ← heap-size[A] – 1

5. MinHeapify(A, 1)

Code Part:

```
#include<iostream>
using namespace std;
void MinHeapify(int A[],int n,int i)
{
   int smallest=i;
   int l=2*i+1;
   int r=2*i+2;
   if(l<n && A[l]<A[smallest])
      smallest=l;
      if(r<n && A[r]<A[smallest])
          smallest=r;
      if(smallest !=i){
          swap(A[i],A[smallest]);
          MinHeapify(A,n,smallest);

   }
}
```

```
void HeapSort(int A[],int n)
{

    for(int i=n;i>=0;i--)
        {
        MinHeapify(A,n,i);
    }
    for(int i=n-1;i>=0;i--)
         {
        swap(A[0],A[i]);
        MinHeapify(A,i,0);
        }
}
void print(int A[],int n)
{
    for(int i=0;i<n;i++)
        cout<<A[i]<<"";
    cout<<" ";

}
int main()
{
    int A[]={11,5,15,2,7};
    int n=sizeof(A)/sizeof(A[0]);
    HeapSort(A,n);
    cout<<"Min Heapsort is";
    print(A,n);
}
```