

Practice assisted project phase 1

1. Write a program in Java to find the fourth smallest element in an unsorted list :

```
2. package main;
3.
4. public class SortedElement {
5.     class MinimumHeap
6.     {
7.         int[] h;
8.         int c;
9.
10.        int heapSize;
11.        int parent(int j)
12.        {
13.            return (j - 1) / 2;
14.        }
15.
16.        int left(int j)
17.        {
18.            return ((2 * j) + 1);
19.        }
20.
21.        int right(int j)
22.        {
23.            return ((2 * j) + 2);
24.        }
25.
26.        int getMin()
27.        {
28.            return h[0];
29.        }
30.
31.        void replaceMax(int y)
32.        {
33.            this.h[0] = y;
34.            heapify(0);
35.        }
36.
37.        MinimumHeap(int arr[], int s)
38.        {
39.            heapSize = s;
```

```

40.         h = arr;
41.         int j = (heapSize - 1) / 2;
42.         while (j >= 0)
43.         {
44.             heapify(j);
45.             j = j - 1;
46.         }
47.     }
48.     int extractMin()
49.     {
50.         if (heapSize == 0)
51.         {
52.             return Integer.MAX_VALUE;
53.         }
54.         int r = h[0];
55.         if (heapSize > 1)
56.         {
57.             h[0] = h[heapSize - 1];
58.             heapify(0);
59.         }
60.         heapSize = heapSize - 1;
61.         return r;
62.     }
63.     void heapify(int j)
64.     {
65.         int lt = left(j);
66.         int rt = right(j);
67.         int minimum = j;
68.         if (lt < heapSize && h[lt] <
h[j])
69.         {
70.             minimum = lt;
71.         }
72.         if (rt < heapSize && h[rt] <
h[minimum] )
73.         {
74.             minimum = rt;
75.         }
76.         if (minimum != j )
77.         {
78.             int t = h[j];
79.             h[j] = h[minimum];
80.             h[minimum] = t;
81.             heapify(minimum);

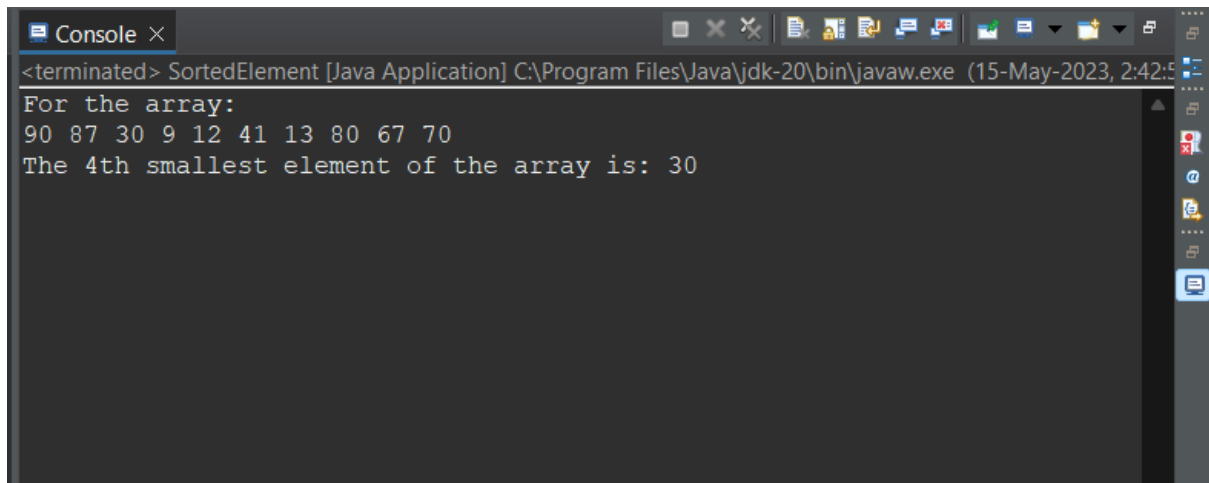
```

```

82.         }
83.     }
84. };
85.     public int findKthSmallest(int a[], int
        s, int k)
86.     {
87.         MinimumHeap mHeap = new
        MinimumHeap(a, s );
88.
89.         for (int j = 0; j < k - 1; j++)
90.         {
91.             mHeap.extractMin();
92.         }
93.
94.         return mHeap.getMin();
95.     }
96.     public static void main(String[] args)
97.     {
98.         SortedElement obj = new
        SortedElement();
99.
100.        int arr1[] = {90, 87, 30, 9, 12, 41,
        13, 80, 67, 70 };
101.
102.        int size = arr1.length;
103.        int k = 4;
104.
105.        System.out.println("For the array:
        ");
106.        for(int i = 0; i < size; i++)
107.        {
108.            System.out.print(arr1[i] + " ");
109.        }
110.
111.        int ele = obj.findKthSmallest(arr1,
        size, k);
112.
113.        System.out.println();
114.        System.out.println("The " + k + "th
        smallest element of the array is: " + ele);
115.
116.    }
117.
118. }

```

OUTPUT-



A screenshot of a Java console window titled "Console". The window shows the output of a Java application named "SortedElement". The output text is as follows:

```
<terminated> SortedElement [Java Application] C:\Program Files\Java\jdk-20\bin\javaw.exe (15-May-2023, 2:42:5...  
For the array:  
90 87 30 9 12 41 13 80 67 70  
The 4th smallest element of the array is: 30
```

The console window has a dark background and a light-colored text. The title bar includes standard Windows window controls (minimize, maximize, close) and a tab labeled "Console". The right side of the window shows a vertical toolbar with various icons for file operations and system functions.