

csvreader.java

```
1 package simple;
2
3 import java.io.FileNotFoundException;
4 import java.io.FileReader;
5 import java.util.Arrays;
6 import java.util.Scanner;
7
8 public class csvreader
9 {
10     public static void main(String args[]) throws FileNotFoundException
11     {
12
13         FileReader file = new FileReader( "C:\\sairam\\masters\\3rd sem Fall_2020\\
14         \\Analysis of algorithm\\project 1\\datafile.csv" );
15         int arr[] = new int [2001];
16         int i=0;
17         try {
18             Scanner input = new Scanner(file);
19             while(input.hasNext())
20             {
21                 arr[i] = input.nextInt();
22                 i++;
23             }
24             input.close();
25         } catch(Exception e)
26         {
27             e.printStackTrace();
28         }
29         median.findMedian(arr);
30     }
31 }
32
33
34 static void findMedian(int arr[])
35 {
36     int median = MedianU(arr,(arr.length)/2 + 1,0,arr.length-1);
37 }
38
39 static int MedianU(int arr[],int k,int x,int y)
40 {
41
42
43     if(x == y)
44     {
45         return arr[x];
46     }
47
48     int m = partition(arr,x,y);
49
50     int length = m - x + 1;
51
52
53     if(length == k)
54     {
55         return arr[m];
56     }
```

```

57
58
59     if(length > k)
60     {
61         return MedianU(arr,k,x,m-1);
62     }
63
64     else
65     {
66         return MedianU(arr,k-length,m+1,y);
67     }
68
69 }
70
71 static int partition(int arr[],int x, int y)
72 {
73
74     int pivotValue = getPivotValue(arr, x, y);
75
76
77     while(x < y)
78     {
79         while(arr[x] < pivotValue)
80         {
81             x ++;
82         }
83
84         while(arr[y] > pivotValue)
85         {
86             y--;
87         }
88
89         if(arr[x] == arr[y])
90         {
91             x ++;
92         }
93         else if(x < y)
94         {
95             int temp = arr[x];
96             arr[x] = arr[y];
97             arr[y] = temp;
98         }
99
100     }
101     return y;
102 }
103
104 static int getPivotValue(int arr[],int x,int y)
105 {
106
107     if(x-y+1 <= 9)
108     {
109         Arrays.sort(arr);
110         return arr[arr.length/2];
111     }
112     int temp[] = null;
113     int medians[] = new int[(int)Math.ceil((double)(y-x+1)/5)];

```

csvreader.java

```

114     int medianIndex = 0;
115
116     while(x <= y)
117     {
118
119         temp = new int[Math.min(5,y-x+1)];
120
121
122         for(int j=0;j<temp.length && x <= y;j++)
123         {
124             temp[j] = arr[x];
125             x++;
126         }
127         Arrays.sort(temp);
128         medians[medianIndex] = temp[temp.length/2];
129         medianIndex++;
130     }
131     return getPivotValue(medians,0,medians.length-1);
132 }
133
134
135 static void printArray(int arr[],int x, int y)
136 {
137     System.out.print("[ ");
138     for(int i=x;i<y;i++)
139     {
140         System.out.print(arr[i] + " ");
141     }
142
143     System.out.println("]");
144 }
145 }
146
147
148

```