MeanMedianDifference.java

```java
package practice;

import java.util.*;

public class MeanMedianDifference {
    public static int[] maxMeanMedian(int input1, int[] input2) {
        Arrays.sort(input2);
        ArrayList<ArrayList<Integer>> outer = new ArrayList<>();
        for (int i = 0; i < (1 << input1); i++) {
            ArrayList<Integer> inner = new ArrayList<>();
            for (int j = 0; j < input1; j++) {
                if ((i & (1 << j)) > 0) {
                    inner.add(input2[j]);
                }
            }
            outer.add(inner);
        }
        double maxDiff = 0;
        int ans[] = null;
        for (ArrayList<Integer> inner1 : outer) {
            if (inner1.size() > 2) {
                double mean = findMean(inner1);
                double median = findMedian(inner1);
                int index = 0;
                if (mean - median > maxDiff) {
                    maxDiff = mean - median;
                    ans = new int[inner1.size()];
                    for (int n : inner1) {
                        ans[index++] = n;
                    }
                }
            }
        }

//      for(ArrayList<Integer> inner1:outer)
//      {
//          if(inner1.size()>2) {
//              double mean=findMean(inner1);
//              double median=findMedian(inner1);
//              int index=0;
//              if(mean-median==maxDiff)
//              {
//                  ans=new int[inner1.size()];
//                  for(int n:inner1) {
//                      ans[index++]=n;
//                  }
//              }
//          }
//      }

//      for (int i = 0; i < ans.length; i++) {
//          System.out.print(ans[i] + " ");
//      }
        return ans;
    }

    public static double findMean(ArrayList<Integer> inner) {
        double sum = 0;
        for (int i = 0; i < inner.size(); i++) {
            sum = sum + inner.get(i);
        }
        return sum / inner.size();
    }

    public static double findMedian(ArrayList<Integer> inner) {
        int n = inner.size();
        if (n % 2 == 1) {
            return inner.get(n / 2);
        } else if (n >= 2) {
            return 0.5 * (inner.get(n / 2) + inner.get((n / 2) - 1));
        } else {
            return 0;
        }
    }

    public static void main(String[] args) {
        int[] arr = { 1, 2, 3, 4 };
        maxMeanMedian(arr.length, arr);
    }
}
```