



Online Java Compiler IDE

For Multiple Files, Custom Library and File Read/Write, use our new - [Advanced Java IDE](#)

```
1 // A simple java program to find maximum score that
2 // maximizing player can get.
3
4 import java.io.*;
5
6 public class GFG {
7
8
9 // Returns the optimal value a maximizer can obtain.
10 // depth is current depth in game tree.
11 // nodeIndex is index of current node in scores[].
12 // isMax is true if current move is of maximizer, else false
13 // scores[] stores leaves of Game tree.
14 // h is maximum height of Game tree
15 static int minimax(int depth, int nodeIndex, boolean isMax,
16                   int scores[], int h)
17 {
18     // Terminating condition. i.e leaf node is reached
19     if (depth == h)
20         return scores[nodeIndex];
21
22     // If current move is maximizer, find the maximum attainable
23     // value
24     if (isMax)
25         return Math.max(minimax(depth+1, nodeIndex*2, false, scores, h),
26                         minimax(depth+1, nodeIndex*2 + 1, false, scores, h));
27
28     // Else (If current move is Minimizer), find the minimum
29     // attainable value
30     else
31         return Math.min(minimax(depth+1, nodeIndex*2, true, scores, h),
32                         minimax(depth+1, nodeIndex*2 + 1, true, scores, h));
33 }
34
35 // A utility function to find Log n in base 2
36 static int log2(int n)
37 {
38     return (n==1)? 0 : 1 + log2(n/2);
39 }
40
41 // Driver code
42
43 public static void main (String[] args) {
44     // The number of elements in scores must be
45     // a power of 2.
46     int scores[] = {3, 5, 2, 9, 12, 5, 23, 23};
47     int n = scores.length;
48     int h = log2(n);
49     int res = minimax(0, 0, true, scores, h);
50     System.out.println( "The optimal value is : " +res);
51 }
52 }
53
54
55 // This code is contributed by vt_m
56
```

Execute Mode, Version, Inputs & Arguments

CommandLine Arguments

Stdin Inputs


Result

CPU Time: 0.10 sec(s), Memory: 33572 kilobyte(s)

compiled and executed in 0.604 sec(s)

```
The optimal value is : 12
```

Note:

1. For file operations - upload files using upload button . Files will be upload to /uploads folder. You can read those files in program from /uploads folder. To write a file from your program, write files to '/myfiles' folder. Please note the uploaded files stored in the server only for the current session.
2. For detailed documentation check - [Our Documentation](#), or check our [Youtube channel](#).

Thanks for using our
Online Java Compiler IDE
to execute your program

**Know Your JDoodle**

- JDoodle Supports 76+ Languages with Multiple Versions and 2 DBs. [Click here](#) to see all.
- Fullscreen - side-by-side code and output is available. click the "⌕" icon near execute button to switch.

JDoodle For Your Organisation

- Do you have any specific compiler requirements?
- Do you want to integrate compilers with your website, webapp, mobile app, courses?

- Dark Theme available. Click on "..."
icon near execute button and select dark
theme.
- You can embed code from JDoodle
directly into your website/blog. [Click here](#)
to know more.
- JDoodle offers an API service. You can
execute programs just by calling our API.
[Click here](#) to know more.
- If you like JDoodle, Please share us in
Social Media. [Click here](#) to share.
- Check our [Documentation Page](#) for
more info.

JDoodle is serving the programming
community since 2013

- Do you need more than our [Embed](#)
and [API](#) features?
- Looking for Multiple Files, Connecting
to DB, Debugging, etc.?
- Are you building any innovative
solution for your students or recruitment?
- Want to run JDoodle in-house?
- Custom Domain, White labelled pages
for your institute?

Contact us - We are happy to help!