```
 1 packageimportstaticimportimportimportimportimportpublicclassPartC_IDSTests
   privatefinalByteArrayOutputStreamoutContent=newByteArrayOutputStream@Test
   publicvoidtestBasicPathfinding()newPrintStreamNodegoal=newNode390null0nullNode
   start=newNode30null0intplanetSize=4Stringfrontier_result="[(3:0)]\n"
   "[(3:315), (3:45), (3:0), (2:0)]\n""[(1:0), (3:90), (2:315), (3:45), (2:45),
   (3:0), (2:0)]\n""(3:0)(3:45)(3:90)\n""4.712\n""7\n"@Testpublicvoid
   testPathOutOfBounds()newPrintStreamNodegoal=newNode190null0nullNodestart=new
   Node30null0intplanetSize=2Stringfrontier_result="fail\n"@Testpublicvoid
   testAdvancedPathfinding()newPrintStreamNodegoal=newNode290null0nullNodestart=
   newNode70null0intplanetSize=8Stringfrontier_result="[(1:0), (7:0), (6:0),
   (5:0), (2:45), (4:0), (3:0), (2:0)]\n""(7:0)(6:0)(5:0)(4:0)(3:0)(2:0)(2:45)
   (2:90)\n""8.142\n""8\n"@TestpublicvoidtestAdvancedPathfinding2()newPrintStream
   Nodegoal=newNode10null0nullNodestart=newNode3180null0intplanetSize=4String
   frontier_result="[(3:180)4.000]\n""5.142\n" Tests;
 2
 3    org.junit.Assert.assertEquals;
 4
 5   java.io.ByteArrayOutputStream;
 6   java.io.PrintStream;
 7
 8   org.junit.Test;
 9
10   Algorithms.PartC_IDS;
11   General.Node;
12
13     {
14          ();
15
16
17        {
18        System.setOut( (outContent));
19
20            (, , , , );
21            (, , , , goal);
22          ;
23
24                  +
25                  +
26                  +
27                  +
28                  + ;
29
30        PartC_IDS.iterativeDeepeningSearch(start, goal, planetSize);
31        assertEquals(frontier_result, outContent.toString());
32     }
33
34
35        {
36        System.setOut( (outContent));
37
38            (, , , , );
39            (, , , , goal);
40          ;
41          ;
42
43        PartC_IDS.iterativeDeepeningSearch(start, goal, planetSize);
44        assertEquals(frontier_result, outContent.toString());
45     }
```

```
46
47
48          {
49           System.setOut( (outContent));
50
51               ( , , , , );
52               ( , , , , goal);
53             ;
54
55                   +
56                   +
57                   + ;
58
59           PartC_IDS.iterativeDeepeningSearch(start, goal, planetSize);
60           assertEquals(frontier_result, outContent.toString());
61      }
62
63
64          {
65           System.setOut( (outContent));
66
67               ( , , , , );
68               ( , , , , goal);
69             ;
70
71                   + ;
72           PartC_IDS.iterativeDeepeningSearch(start, goal, planetSize);
73           assertEquals(frontier_result, outContent.toString());
74      }
75 }
76
```