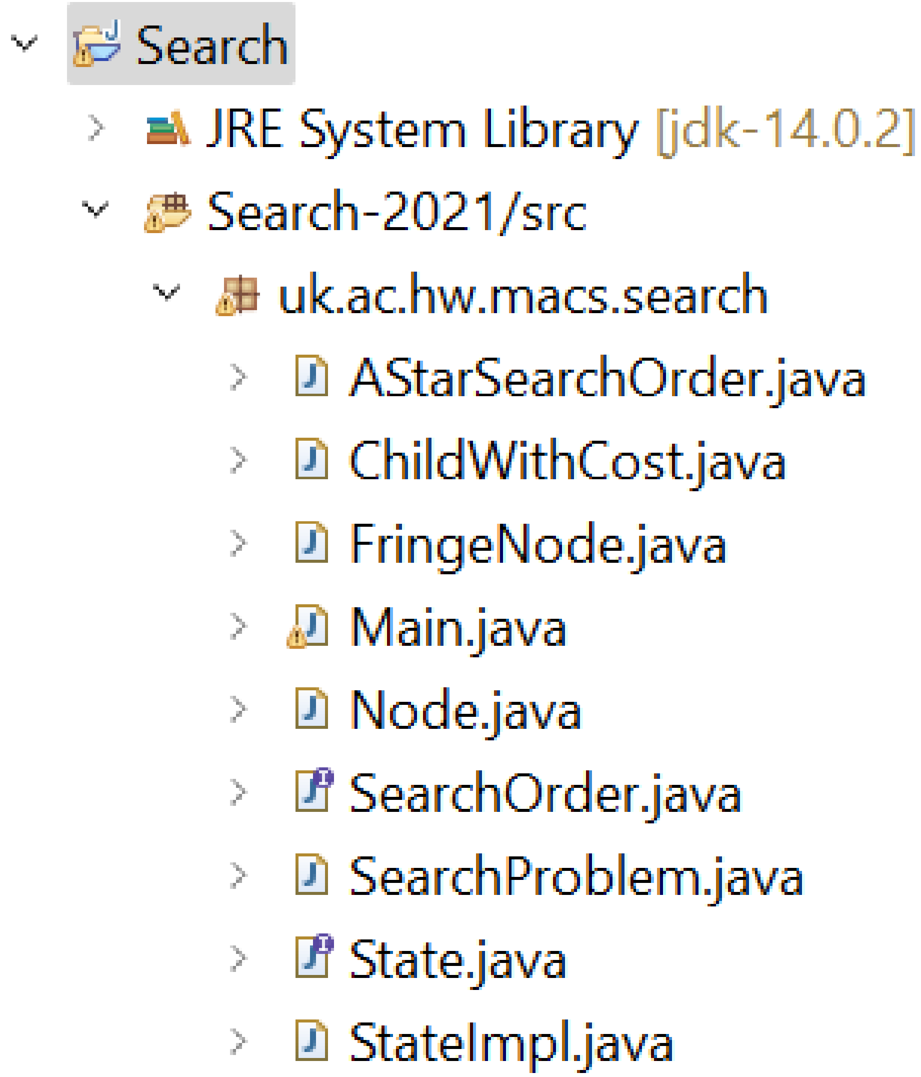
AMAZON CODING CHALLENGE BRIGHT NETWORK INTERNSHIP

Files and Output

SOURCE FILES



```
Search
  JRE System Library [jdk-14.0.2]
  Search-2021/src
    uk.ac.hw.macs.search
      AStarSearchOrder.java
      ChildWithCost.java
      FringeNode.java
      Main.java
      Node.java
      SearchOrder.java
      SearchProblem.java
      State.java
      StateImpl.java
```

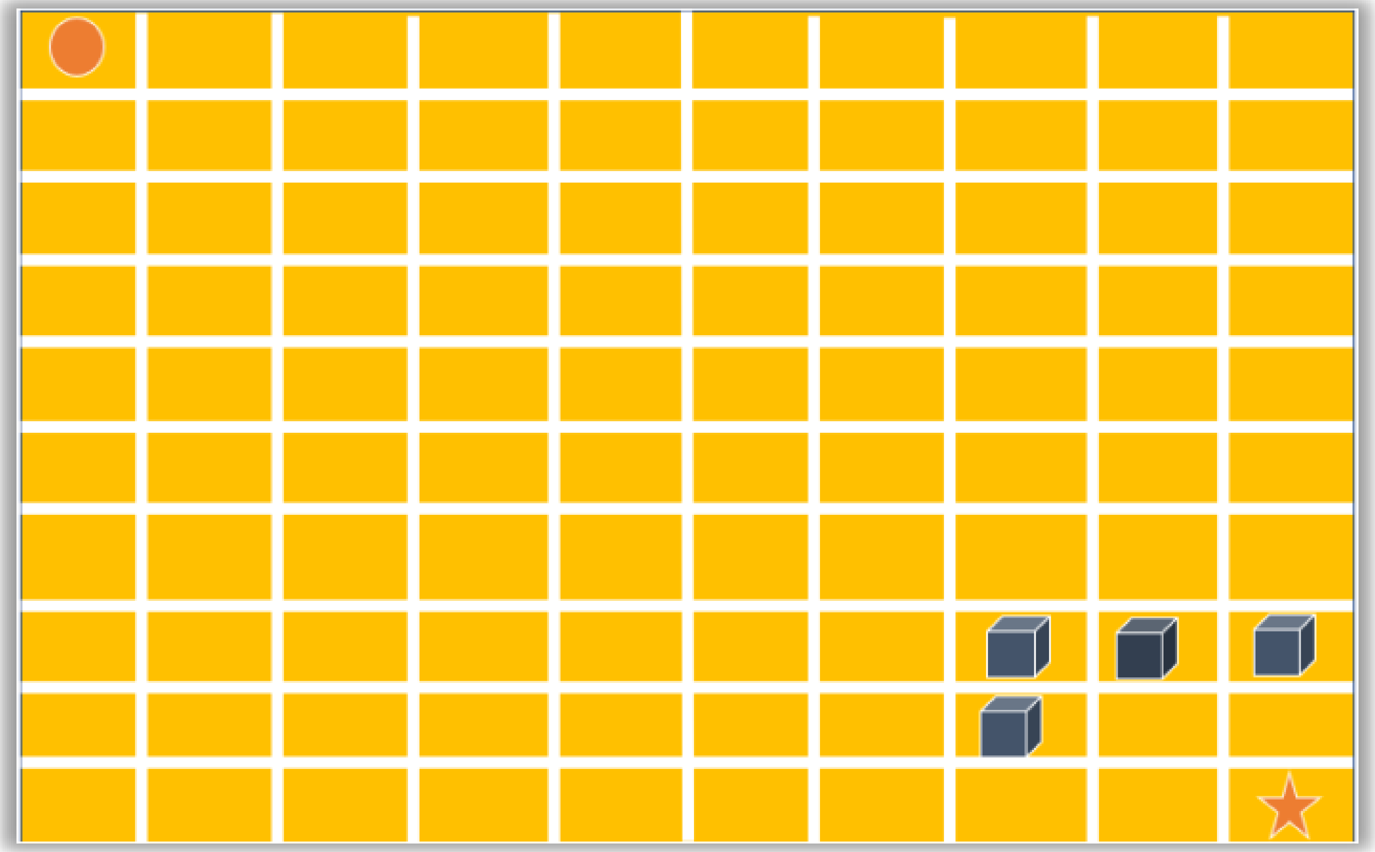
FILES DESCRIPTION

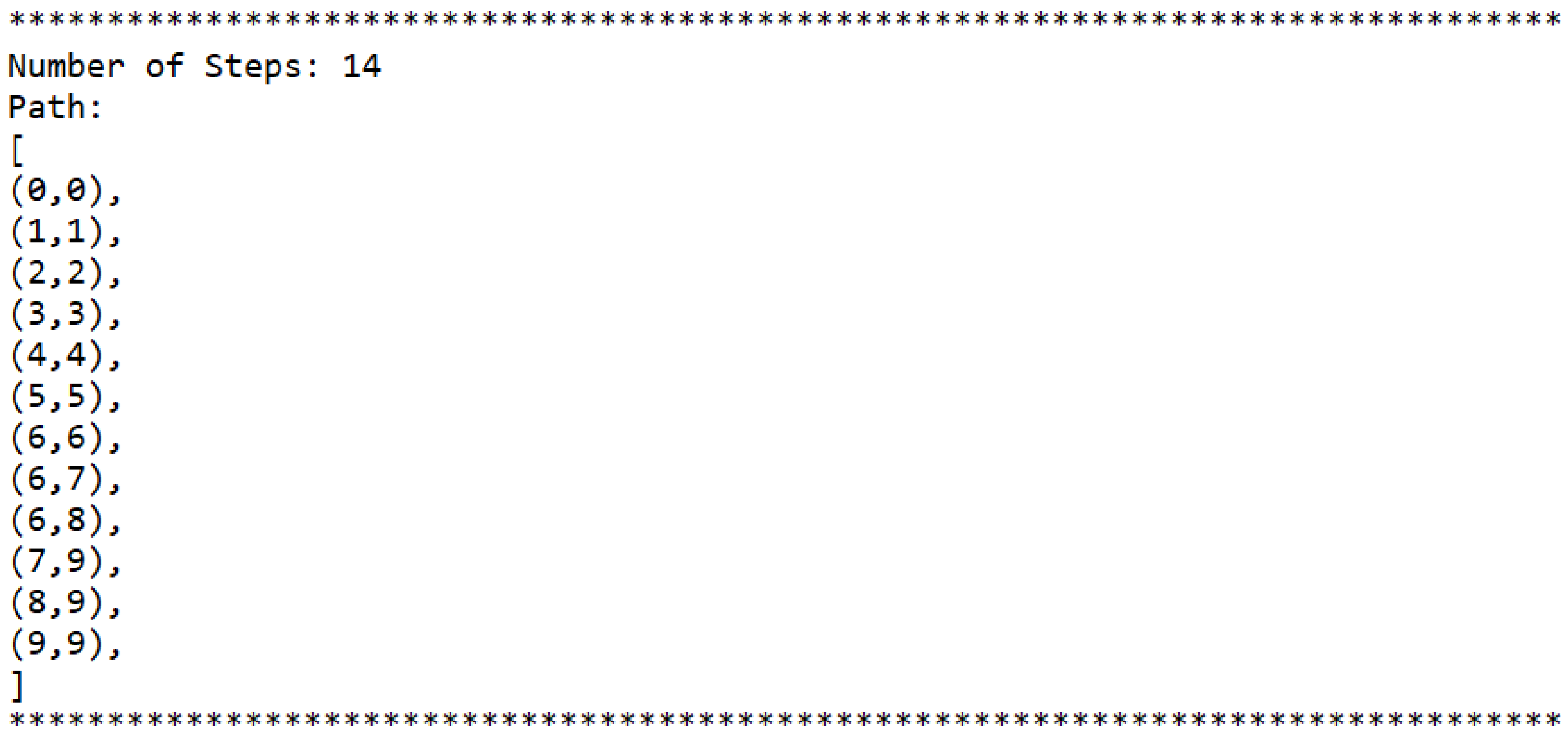
- SearchOrder- Encodes a search order by describing how nodes are added to the fringe. Note that nodes are always removed from the front of the fringe.
- AStarSearchOrder- Class that contains methods to add nodes to a fringe during A* Search. The states are expanded in the increasing order of the total cost. No state can be expanded more than once and when inserting nodes to the fringe, ties are resolved so that states with the lowest state value are ordered first.
- ChildWithCost- Represents a connection in the state graph: a node, and the cost of getting from the parent to this node.
- FringeNode- Represents a node on the frontier of the search space. Includes the actual node, the parent node (i.e., the node that was expanded to add this one), as well as the cost of getting to this node (the "g" value).

FILES DESCRIPTION

- Node- Represents a single node in the search space: it has a value, as well as a set of child nodes with associated costs on the transitions.
- State- Represents a state in the search space.
- StateImpl- Class representing a state in the grid, implements the State interface
- SearchProblem- Represents a search problem, by running the given search order on given search spaces.
- Main- Class to represent the given grids and run the A* Search Algorithm on them

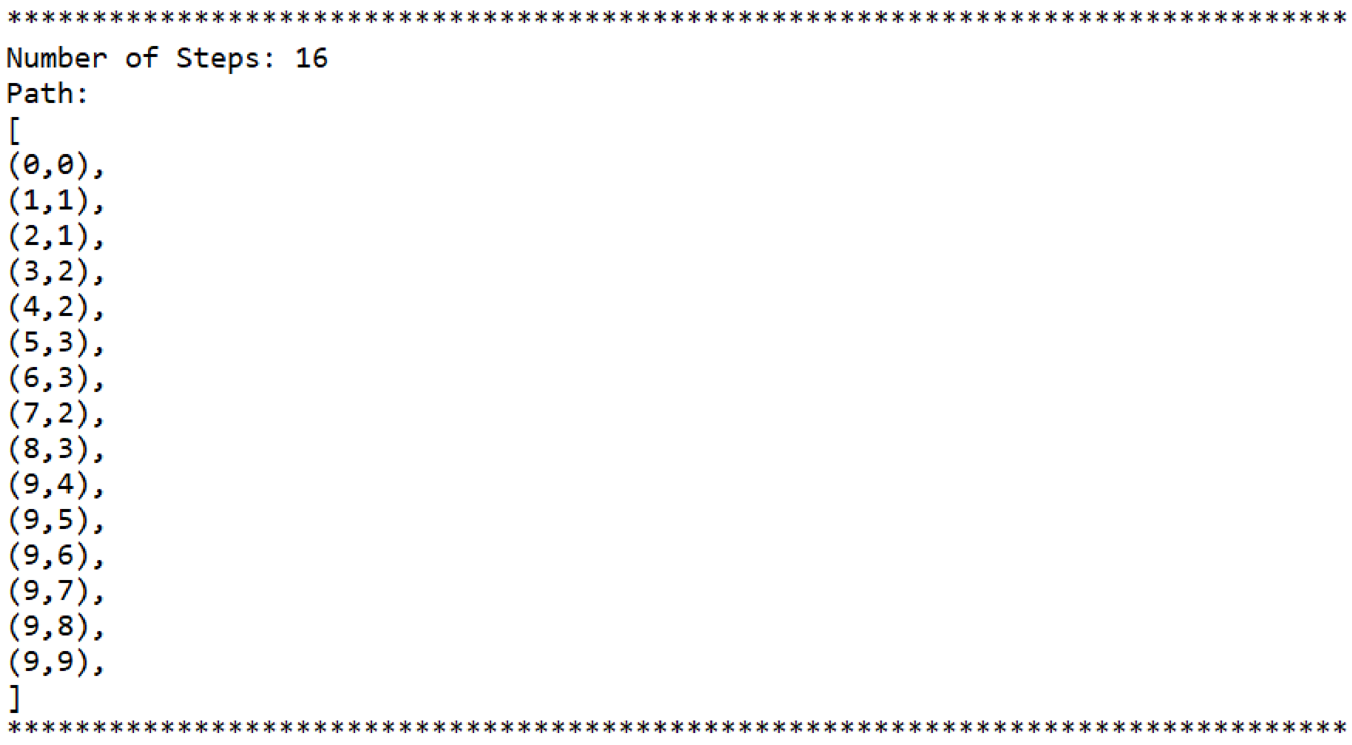
PHASE-I GRID





```
********************************************************************************
Number of Steps: 14
Path:
[
(0,0),
(1,1),
(2,2),
(3,3),
(4,4),
(5,5),
(6,6),
(6,7),
(6,8),
(7,9),
(8,9),
(9,9),
]
********************************************************************************
```

OUTPUT- PHASE I



```
********************************************************************************
Number of Steps: 16
Path:
[
(0,0),
(1,1),
(2,1),
(3,2),
(4,2),
(5,3),
(6,3),
(7,2),
(8,3),
(9,4),
(9,5),
(9,6),
(9,7),
(9,8),
(9,9),
]
********************************************************************************
```

OUTPUT PHASE-2

FOR ANY CLARIFICATIONS,

please contact
shreyakala@ymail.com