| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-all.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | | |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)   [**NEXT CLASS**](http://docs.google.com/info/gridworld/grid/Location.html) | [**FRAMES**](http://docs.google.com/index.html?info/gridworld/grid/Grid.html)    [**NO FRAMES**](http://docs.google.com/Grid.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **info.gridworld.grid**

Interface Grid<E>

**All Known Implementing Classes:** [AbstractGrid](http://docs.google.com/info/gridworld/grid/AbstractGrid.html), [BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html), [UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)

public interface **Grid<E>**

Grid provides an interface for a two-dimensional, grid-like environment containing arbitrary objects.

This interface is testable on the AP CS A and AB exams.

| **Method Summary** | |
| --- | --- |
| [E](http://docs.google.com/info/gridworld/grid/Grid.html) | [**get**](http://docs.google.com/info/gridworld/grid/Grid.html#get(info.gridworld.grid.Location))([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)            Returns the object at a given location in this grid. |
| [ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[Location](http://docs.google.com/info/gridworld/grid/Location.html)> | [**getEmptyAdjacentLocations**](http://docs.google.com/info/gridworld/grid/Grid.html#getEmptyAdjacentLocations(info.gridworld.grid.Location))([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)            Gets the valid empty locations adjacent to a given location in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest). |
| [ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[E](http://docs.google.com/info/gridworld/grid/Grid.html)> | [**getNeighbors**](http://docs.google.com/info/gridworld/grid/Grid.html#getNeighbors(info.gridworld.grid.Location))([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)            Gets the neighboring occupants in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest). |
| int | [**getNumCols**](http://docs.google.com/info/gridworld/grid/Grid.html#getNumCols())()            Returns the number of columns in this grid. |
| int | [**getNumRows**](http://docs.google.com/info/gridworld/grid/Grid.html#getNumRows())()            Returns the number of rows in this grid. |
| [ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[Location](http://docs.google.com/info/gridworld/grid/Location.html)> | [**getOccupiedAdjacentLocations**](http://docs.google.com/info/gridworld/grid/Grid.html#getOccupiedAdjacentLocations(info.gridworld.grid.Location))([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)            Gets the valid occupied locations adjacent to a given location in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest). |
| [ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[Location](http://docs.google.com/info/gridworld/grid/Location.html)> | [**getOccupiedLocations**](http://docs.google.com/info/gridworld/grid/Grid.html#getOccupiedLocations())()            Gets the locations in this grid that contain objects. |
| [ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[Location](http://docs.google.com/info/gridworld/grid/Location.html)> | [**getValidAdjacentLocations**](http://docs.google.com/info/gridworld/grid/Grid.html#getValidAdjacentLocations(info.gridworld.grid.Location))([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)            Gets the valid locations adjacent to a given location in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest). |
| boolean | [**isValid**](http://docs.google.com/info/gridworld/grid/Grid.html#isValid(info.gridworld.grid.Location))([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)            Checks whether a location is valid in this grid. |
| [E](http://docs.google.com/info/gridworld/grid/Grid.html) | [**put**](http://docs.google.com/info/gridworld/grid/Grid.html#put(info.gridworld.grid.Location,%20E))([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc, [E](http://docs.google.com/info/gridworld/grid/Grid.html) obj)            Puts an object at a given location in this grid. |
| [E](http://docs.google.com/info/gridworld/grid/Grid.html) | [**remove**](http://docs.google.com/info/gridworld/grid/Grid.html#remove(info.gridworld.grid.Location))([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)            Removes the object at a given location from this grid. |

| **Method Detail** |
| --- |

### getNumRows

int **getNumRows**()

Returns the number of rows in this grid.

**Returns:**the number of rows, or -1 if this grid is unbounded

### getNumCols

int **getNumCols**()

Returns the number of columns in this grid.

**Returns:**the number of columns, or -1 if this grid is unbounded

### isValid

boolean **isValid**([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)

Checks whether a location is valid in this grid.

Precondition: loc is not null

**Parameters:**loc - the location to check **Returns:**true if loc is valid in this grid, false otherwise

### put

[E](http://docs.google.com/info/gridworld/grid/Grid.html) **put**([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc,  
 [E](http://docs.google.com/info/gridworld/grid/Grid.html) obj)

Puts an object at a given location in this grid.

Precondition: (1) loc is valid in this grid (2) obj is not null

**Parameters:**loc - the location at which to put the objectobj - the new object to be added **Returns:**the object previously at loc (or null if the location was previously unoccupied)

### remove

[E](http://docs.google.com/info/gridworld/grid/Grid.html) **remove**([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)

Removes the object at a given location from this grid.

Precondition: loc is valid in this grid

**Parameters:**loc - the location of the object that is to be removed **Returns:**the object that was removed (or null if the location is unoccupied)

### get

[E](http://docs.google.com/info/gridworld/grid/Grid.html) **get**([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)

Returns the object at a given location in this grid.

Precondition: loc is valid in this grid

**Parameters:**loc - a location in this grid **Returns:**the object at location loc (or null if the location is unoccupied)

### getOccupiedLocations

[ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[Location](http://docs.google.com/info/gridworld/grid/Location.html)> **getOccupiedLocations**()

Gets the locations in this grid that contain objects.

**Returns:**an array list of all occupied locations in this grid

### getValidAdjacentLocations

[ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[Location](http://docs.google.com/info/gridworld/grid/Location.html)> **getValidAdjacentLocations**([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)

Gets the valid locations adjacent to a given location in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest).

Precondition: loc is valid in this grid

**Parameters:**loc - a location in this grid **Returns:**an array list of the valid locations adjacent to loc in this grid

### getEmptyAdjacentLocations

[ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[Location](http://docs.google.com/info/gridworld/grid/Location.html)> **getEmptyAdjacentLocations**([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)

Gets the valid empty locations adjacent to a given location in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest).

Precondition: loc is valid in this grid

**Parameters:**loc - a location in this grid **Returns:**an array list of the valid empty locations adjacent to loc in this grid

### getOccupiedAdjacentLocations

[ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[Location](http://docs.google.com/info/gridworld/grid/Location.html)> **getOccupiedAdjacentLocations**([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)

Gets the valid occupied locations adjacent to a given location in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest).

Precondition: loc is valid in this grid

**Parameters:**loc - a location in this grid **Returns:**an array list of the valid occupied locations adjacent to loc in this grid

### getNeighbors

[ArrayList](http://java.sun.com/j2se/1.5.0/docs/api/java/util/ArrayList.html?is-external=true)<[E](http://docs.google.com/info/gridworld/grid/Grid.html)> **getNeighbors**([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)

Gets the neighboring occupants in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest).

Precondition: loc is valid in this grid

**Parameters:**loc - a location in this grid **Returns:**returns an array list of the objects in the occupied locations adjacent to loc in this grid

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-all.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | | |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)   [**NEXT CLASS**](http://docs.google.com/info/gridworld/grid/Location.html) | [**FRAMES**](http://docs.google.com/index.html?info/gridworld/grid/Grid.html)    [**NO FRAMES**](http://docs.google.com/Grid.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |