| | [**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   [**NEXT CLASS**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html) | [**FRAMES**](http://docs.google.com/index.html?info/gridworld/grid/AbstractGrid.html)    [**NO FRAMES**](http://docs.google.com/AbstractGrid.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#2s8eyo1) |

## **info.gridworld.grid**

Class AbstractGrid<E>

[java.lang.Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true)  
 **info.gridworld.grid.AbstractGrid<E>**

**All Implemented Interfaces:** [Grid](http://docs.google.com/info/gridworld/grid/Grid.html)<E> **Direct Known Subclasses:** [BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html), [UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)

public abstract class **AbstractGrid<E>**extends [Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true)implements [Grid](http://docs.google.com/info/gridworld/grid/Grid.html)<E>

AbstractGrid contains the methods that are common to grid implementations.

The implementation of this class is testable on the AP CS AB exam.

| **Constructor Summary** | |
| --- | --- |
| [**AbstractGrid**](http://docs.google.com/info/gridworld/grid/AbstractGrid.html#AbstractGrid())() |

| **Method Summary** | |
| --- | --- |
| [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/AbstractGrid.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/AbstractGrid.html)> | [**getNeighbors**](http://docs.google.com/info/gridworld/grid/AbstractGrid.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). |
| [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/AbstractGrid.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)> | [**getValidAdjacentLocations**](http://docs.google.com/info/gridworld/grid/AbstractGrid.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). |
| [String](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/String.html?is-external=true) | [**toString**](http://docs.google.com/info/gridworld/grid/AbstractGrid.html#toString())()            Creates a string that describes this grid. |

| **Methods inherited from class java.lang.**[**Object**](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) |
| --- |
| [clone](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#clone()), [equals](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#equals(java.lang.Object)), [finalize](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#finalize()), [getClass](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#getClass()), [hashCode](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#hashCode()), [notify](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#notify()), [notifyAll](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#notifyAll()), [wait](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#wait()), [wait](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#wait(long)), [wait](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#wait(long,%20int)) |

| **Methods inherited from interface info.gridworld.grid.**[**Grid**](http://docs.google.com/info/gridworld/grid/Grid.html) |
| --- |
| [get](http://docs.google.com/info/gridworld/grid/Grid.html#get(info.gridworld.grid.Location)), [getNumCols](http://docs.google.com/info/gridworld/grid/Grid.html#getNumCols()), [getNumRows](http://docs.google.com/info/gridworld/grid/Grid.html#getNumRows()), [getOccupiedLocations](http://docs.google.com/info/gridworld/grid/Grid.html#getOccupiedLocations()), [isValid](http://docs.google.com/info/gridworld/grid/Grid.html#isValid(info.gridworld.grid.Location)), [put](http://docs.google.com/info/gridworld/grid/Grid.html#put(info.gridworld.grid.Location,%20E)), [remove](http://docs.google.com/info/gridworld/grid/Grid.html#remove(info.gridworld.grid.Location)) |

| **Constructor Detail** |
| --- |

### AbstractGrid

public **AbstractGrid**()

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

### getNeighbors

public [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/AbstractGrid.html)> **getNeighbors**([Location](http://docs.google.com/info/gridworld/grid/Location.html) loc)

**Description copied from interface:** [**Grid**](http://docs.google.com/info/gridworld/grid/Grid.html#getNeighbors(info.gridworld.grid.Location)) 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

**Specified by:**[getNeighbors](http://docs.google.com/info/gridworld/grid/Grid.html#getNeighbors(info.gridworld.grid.Location)) in interface [Grid](http://docs.google.com/info/gridworld/grid/Grid.html)<[E](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)> **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

### getValidAdjacentLocations

public [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)

**Description copied from interface:** [**Grid**](http://docs.google.com/info/gridworld/grid/Grid.html#getValidAdjacentLocations(info.gridworld.grid.Location)) 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

**Specified by:**[getValidAdjacentLocations](http://docs.google.com/info/gridworld/grid/Grid.html#getValidAdjacentLocations(info.gridworld.grid.Location)) in interface [Grid](http://docs.google.com/info/gridworld/grid/Grid.html)<[E](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)> **Parameters:**loc - a location in this grid **Returns:**an array list of the valid locations adjacent to loc in this grid

### getEmptyAdjacentLocations

public [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)

**Description copied from interface:** [**Grid**](http://docs.google.com/info/gridworld/grid/Grid.html#getEmptyAdjacentLocations(info.gridworld.grid.Location)) 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

**Specified by:**[getEmptyAdjacentLocations](http://docs.google.com/info/gridworld/grid/Grid.html#getEmptyAdjacentLocations(info.gridworld.grid.Location)) in interface [Grid](http://docs.google.com/info/gridworld/grid/Grid.html)<[E](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)> **Parameters:**loc - a location in this grid **Returns:**an array list of the valid empty locations adjacent to loc in this grid

### getOccupiedAdjacentLocations

public [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)

**Description copied from interface:** [**Grid**](http://docs.google.com/info/gridworld/grid/Grid.html#getOccupiedAdjacentLocations(info.gridworld.grid.Location)) 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

**Specified by:**[getOccupiedAdjacentLocations](http://docs.google.com/info/gridworld/grid/Grid.html#getOccupiedAdjacentLocations(info.gridworld.grid.Location)) in interface [Grid](http://docs.google.com/info/gridworld/grid/Grid.html)<[E](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)> **Parameters:**loc - a location in this grid **Returns:**an array list of the valid occupied locations adjacent to loc in this grid

### toString

public [String](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/String.html?is-external=true) **toString**()

Creates a string that describes this grid.

**Overrides:**[toString](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#toString()) in class [Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) **Returns:**a string with descriptions of all objects in this grid (not necessarily in any particular order), in the format {loc=obj, loc=obj, ...}

| | [**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   [**NEXT CLASS**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html) | [**FRAMES**](http://docs.google.com/index.html?info/gridworld/grid/AbstractGrid.html)    [**NO FRAMES**](http://docs.google.com/AbstractGrid.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#2s8eyo1) |