Use the [GridWorld Quick Reference](https://secure-media.collegeboard.org/apc/ap_comp_sci_a_quick_reference.pdf) if needed.Test the code out during and after each step to make sure it is error free

**Overview:**

Santong island is an uninhabited speck of land in the South Pacific Ocean. In the late 1980’s, naturalist Jessica Tong discovered two previously unknown species that are unique to this island. One is the jeroo, a rare mammal similar to the wallabies of Australia. The other is the large winsum flower that is the jeroo’s primary food source.

**~~~~~~~~~~~~~~~~ JerooRunner.java ~~~~~~~~~~~~~~~~**

**Part 0:**

Create a new Class named JerooRunner w/ a main method.

Import info.gridworld.actor.\* and info.gridworld.grid.\*

**~~~~~~~~~~~~~~~~ SantongIslandWorld.java ~~~~~~~~~~~~~~~~**

**Part 1:**

Create a new Class named SantongIslandWorld that inherits ActorWorld.

Write a default constructor for the SantongIslandWorld that uses the ActorWorld(Grid<Actor>) constructor.

The default SantongIslandWorld should be 24 x 24.

New Classes/Methods to use:

new BoundedGrid<Actor>( int rows, int cols )

**\*\* Test Part 1 out before moving on to part 2 \*\***

**Part 2:**

In the JerooRunner, make a default SantongIslandWorld named island and show the island.

It should show up as a 24 x 24 grid.

New Classes/Methods to use:

new SantongIslandWorld()

island.show()

**\*\* Test Part 2 out before moving on to part 3 \*\***

**~~~~~~~~~~~~~~~~ WinsumFlower.java ~~~~~~~~~~~~~~~~**

**Part 3:**

Create a new Class named WinsumFlower that inherits Flower

The default WinsumFlower needs to be yellow, then you will override the act method in flower to keep the flower yellow instead of dying.

New Classes/Methods to use:

super( Color color )

**\*\* Test part 3 out before moving on to part 4 \*\***

**Part 4:**

In the JerooRunner, you need to add 10 new WinsumFlowers to the island in random places. While you are doing that, go ahead and add 5 new Rocks to random locations as well.

New Classes/Methods to use:

new WinsumFlower(), new Rock() or new Rock( Color color )

**\*\* Test part 4 out before moving on to part 5 \*\***

**~~~~~~~~~~~~~~~~ JerooActor.java ~~~~~~~~~~~~~~~~**

**Part 5:**

Create a new class named JerooActor, it needs an instance variable that is a list of WinsumFlowers.

Create a constructor that takes 2 int parameters >> JerooActor( int comp\_dir, int flowers)

In this constructor, change it's direction that it is facing, then add the correct amount of winsum flowers.

New Classes/Methods to use:

ArrayList<WinsumFlower>, use a loop to add flowers to the list

setDirection( comp\_dir )

**\*\* Test part 5 out before moving on to part 6\*\***

**Part 6:**

In the JerooRunner, add 4 new JerooActors to the world, facing in different directions and having a different amount of flowers. Use the [GridWorld Quick Reference](https://secure-media.collegeboard.org/apc/ap_comp_sci_a_quick_reference.pdf), go to Location Compass Directions.

New Classes/Methods to use:

New JerooActor( int comp\_dir, int flowers )

**\*\* Test part 6 out before moving on to part 7\*\***

**Part 7:**

In the JerooActor class override the Actor act method. Then you will need to write the following void methods and call them in the act method to make sure they work. Edit the island by adding rocks, Winsum Flowers, or other Jeroos so you can test the code.

| **Methods** | **Description** | **Examples** |
| --- | --- | --- |
| hop()  hop(int num) | Moves in the direction you are facing. | hop();  hop( 2 ); |
| turn( int rel\_dir) | Turns in place in the direction given. | turn( Location.RIGHT );  turn( Location.LEFT); |
| pick() | Picks up a flower at the current location. | if( isFlower( getLocation() )  pick(); |
| plant() | Plants a flower at the current location. | plant(); |
| toss() | Tosses a flower in the direction you are facing and crushes a rock if one is there. | if( isRock( nextLoc ) )  toss(); |
| canMove( Location loc ) | Returns true if you can move in the direction given, otherwise it returns false. | canMove( getLocation().getAdjacentLocation( int comp\_dir) ) |
| isRock( Location loc ) | Returns true if there is a rock in the direction given, otherwise it returns false. | isRock( getLocation().getAdjacentLocation( int comp\_dir) ) |
| isFlower( Location loc ) | Returns true if there is a flower in the direction given, otherwise it returns false. | isFlower( getLocation() );  isFlower( getLocation().getAdjacentLocation( int comp\_dir ) ) |

**\*\* Test all of part 7 out before turning it in \*\***