**Jeroo Reference Sheet**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Action Methods** | **Boolean Methods** | **Logical Operators** | **Relative Directions** | **Compass Directions** |
| hop() | isWater(relative direction) | **!** - not | LEFT | NORTH |
| hop(n) | isNet(relative direction) | **&&** - and | RIGHT | SOUTH |
| pick() | isClear(relative direction) | **||** - or | AHEAD | EAST |
| plant() | isJeroo(relative direction) |  | HERE | WEST |
| toss() | isFlower(relative direction) |  |  |  |
| turn(direction) | hasFlower() |  |  |  |
| give(direction) | **isFacing(compass direction)** |  |  |  |

**Control Structures**

|  |  |
| --- | --- |
| **Control Structure** | **Example** |
| while( **condition** ) {  } | while(**!**bob.isWater(AHEAD)) // not water ahead  {  bob.hop();  } |
| if( **condition** ) {  } | if(bob.isNet(AHEAD))  {  bob.toss();  bob.hop();  } |
| if( **condition** ) // if\_else {  } else {  } | if(bob.isFlower(HERE))  {  bob.pick();  }  else  {  bob.hop();  } |

|  |  |
| --- | --- |
| **Control Structure** | **Example** |
| if( **condition** ) // cascading if {  }  else if( **condition** )  {  }  else  {  } | if(bob.isNet(AHEAD))  {  bob.toss();  }  else if(bob.isFlower(AHEAD)) {  bob.hop();  bob.pick();  }  else  {  bob.hop(); } |
| if( **condition** && **condition** ) // and {  } | if(bob.isFacing(SOUTH) && bob.isWater(AHEAD))  {  bob.turn(LEFT);  bob.turn(LEFt);  } |
| if( **condition** || **condition** ) // or {  } else {  } | if(bob.isWater(AHEAD)) || bob.isJeroo(AHEAD))  {  bob.turn(RIGHT);  bob.hop(8);  }  else  {  bob.hop();  } |