

1  
2  
3

# GridWorld Role Play

## Guidelines for Teachers

---

4 **I. “Actors” Needed**

- 5     **1. RolePlayBugRunner**  
6     **2. Grid\*\***  
7     **3. ActorWorld**  
8     **4. Bug1**  
9     **5. Flower1 (added after Bug1 acts)**  
10    **6. Bug2**  
11    **7. Flower2 (added after Bug2 acts)**  
12    **8. Flower**  
13    **9. Rock**  
14    **10. Critters (1-3 Critters for second simulation)**

15  
16 **II. Use GridWorld Simulation for display**

17  
18 **III. Use nerfball to throw around room to show “active” object. Post-its may be**  
19 **posted to the ball to pass parameters. Mini-white boards may also be used to**  
20 **show the “active” object as well as the place where parameters are written**  
21

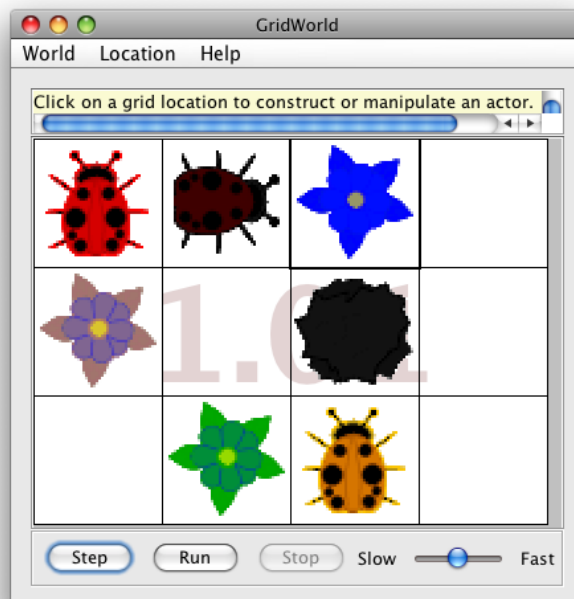
22 **IV. RC Cola! Remember for 2D Arrays, the elements are referred to in**  
23 **[row][column] (aka ‘row-major” or left-right and top-down) notation.**  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44

- 45 **V. Schedule**
- 46 **a. Day 1**
- 47 **i. GW Overview**
- 48 **i. GW Role Play is an approximation to code, not exact...few**
- 49 **getters/setters.**
- 50 **ii. Follow GW Role Play rules (be the computer!)**
- 51 **iii. Use visible grid (Excel, whiteboard, GridWorld Simulation, etc.)**
- 52 **and choose strategic placement of Actors to investigate “special**
- 53 **cases (see below).**
- 54 **iv. Assign appropriate roles. Use understudies for large class sizes.**
- 55
- 56 **b. Day 2**
- 57 **i. Object construction – get through constructing on this day. A**
- 58 **“test run” of the role play.**
- 59 **ii. For the BugRunner role play, try to include in your initial setup a**
- 60 **situation that includes the following cases:**
- 61 **a. A Bug that attempts to move off the Grid**
- 62 **b. A Bug that attempts to move into a location occupied by a**
- 63 **Flower**
- 64 **c. A Bug that attempts to move into a location occupied by a**
- 65 **Rock**
- 66 **d. A Bug that attempts to move into a location occupied by**
- 67 **another Bug.**
- 68

69 **Below is an example of a 3 x 4 grid with some special cases. I usually use a**

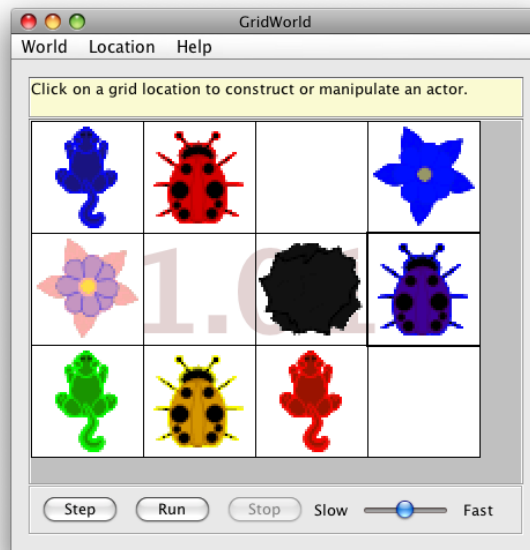
70 **5x5 grid in my classroom for a larger grid area to place the “Actors.”**

71



73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87

- b. **Day 3 - BugRunner**
  - i. **Run through ActorWorld “step” method one time using BugRunner. This a good time for students to see the “algorithmic” nature of the role play the “rules” the computer has to follow in order for the “game” to run.**
  
- d. **Day 4 CritterRunner**
  - i. **For CritterRunner role play, I bypass the “Object Construction” phase in the interest of time; I simply give students role and place them in the appropriate seats in the classroom.**
  - ii. **Again, strategically place Critters, Bug, Flower, and Rocks to cover situations where the Actors attempt to move off the board, move into a location that is occupied, etc. Below is a sample setup:**



88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102

- d. **5-6 Code Walkthrough**
  - i. **Review syntax...all conceptual questions/design decisions have been discussed.**
  - ii. **Solicit ideas of modifications to GW (“toral” world, new Critters**

**This version of the GridWorld role play has been edited with input from participants from the:  
AP Summer Institute at Cal State San Marcos 2007, 2008, 2009, and 2010  
Silver State AP Summer Institute at Las Vegas 2007, 2008, and 2010  
AP By the Sea 2010  
Loyola High School AP Computer Science students spring 2008, 2009, 2010, and 2011**

## You are the RolePlayBugRunner (“The Producer”)

---

### Run

When the role play is set to begin (this role play assumes a 5 x 5 board)

- 1) Construct an ActorWorld by picking the pre-selected person in the room and saying, <person name>, “**Construct** yourself as an ActorWorld”
- 2) Construct a Bug by picking the pre-selected person in the room and saying, <person name>, “**Construct** yourself as an Bug; your color is Red”
- 3) Ask your Bug to set its direction to East. (This is not in the role play script...Bug, you can just change your direction)
- 4) Ask your ActorWorld to **add** Bug <person name> at position (3,3)
- 5) Construct a Bug by picking pre-selected person in the room and saying, <person name>, “**Construct** yourself as an Bug; your color is “Blue”
- 6) Ask your Bug to set its direction to North. (This is not in the role play script...Bug, you can just change your direction)
- 7) Ask your ActorWorld to **add** Bug <person name> at position (0,0)
- 8) Construct a Flower by picking the pre-selected person in the room and saying, <person name>, “**Construct** yourself as a Flower”
- 9) Ask your ActorWorld to **add** Flower <person name> at position (3,4)
- 10) Construct a Rock by picking the pre-selected in the room and saying, <person name>, “**Construct** yourself as an Rock”
- 11) Ask your ActorWorld to **add** Rock <person name> at position (2,3)
- 12) Have a **commercialInterruption** (on the next page)
- 13) Ask your ActorWorld to **step**.
- 14) Have a **commercialInterruption**
- 15) Ask your ActorWorld to **step**.
- 16) Have a **commercialInterruption**

147 **commercialInterruption**

148

149 To process a **commercialInterruption**

150

151 1) Narrate “We now interrupt this role play for a commercial interruption. The  
152 sponsors hope that you will pay attention to the display during this break.”

153 2) For each Actor (e.g. Bug, Flower, Rock, Critter – NOT ActorWorld or Grid) who  
154 is currently involved in the role play, do the following:

155 a) Ask the actor if their Grid is null

156 i) If the actor says, “yes”, then skip to the next actor.

157 b) Ask the actor for his/her location.

158 c) Ask the actor for his/her color.

159 d) Ask the actor for his/her direction.

160 e) Update whatever display your instructor has provided (e.g. blackboard,  
161 whiteboard, poster, etc.) to reflect this actor’s status in the world.

162 3) Narrate “No more actors. We now resume our regularly scheduled program.”  
163 (Feel free to emphasize the last word...)

164

164 **ActorWorld** \_\_\_\_\_ (*your name here*)

165

166

167 **Private information**

168

169                      **your Grid:** \_\_\_\_\_

170

171

172 **Actors List (in order of appearance)**

173

174 ActorWorld: \_\_\_\_\_

175 Grid: \_\_\_\_\_

176 Bug: \_\_\_\_\_

177 Flower: \_\_\_\_\_

178 Rock: \_\_\_\_\_

179

# You are an ActorWorld. (“The Director”)

---

## Constructing Yourself

When you are constructed you may be given:

- Your **Grid**

- 1) If you are not given this information:
    - a) Construct a Grid by picking an unused person in the room and saying, <person name>, “**Construct** yourself as a Grid”
    - 2) Remember the name of your Grid by recording it in the appropriate place on your **Private Data** sheet.
    - 3) Say “Done constructing ActorWorld <your name>.”
- 

## add

When asked to **add**, you will be given an Actor and a Location

- 1) Ask <the actor you are given> to **putSelfInGrid** <your grid> at <the location you were given>
  - 2) Say “<name of the actor> inserted from perspective of ActorWorld.”
- 

## step

When asked to **step**

- 1) Ask <your Grid> to **getOccupiedLocations** and place the occupied locations into a list (use row, column format)
- 2) For each Location in the List:
  - a) Ask the Grid for **nameOfActorAt** each location in the list passed you from Grid. (there is not script for this, Grid will simply return the list of names.
  - b) Add that name to your <listOfNames>
- 3) For each name in your <listOfNames>
  - a) Ask that person for the name of their Grid
  - b) If they do not say, “blank” (meaning null)
    - i) Ask that person to **act**
- 4) Say “Done stepping.”

listOfNames

219  
220  
221  
222

**Grid** \_\_\_\_\_ (your name here)

**Private information (aka “who is where”)**

	0	1	2	3	4	5	6	7	8	9
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

223

223 **You are a Grid. (“The Stage”)**

224

225

---

226 **Constructing Yourself**

227

228 1) Verify that the table you have (on paper) is currently empty.

229 2) Say “Done constructing Grid <your name>.”

230

231

232

---

233 **isAValidLocation**

234

235 When asked **isAValidLocation**, you will be given a Location

236 :

237 1) If the value of row in that location is between 0 and 4 inclusive AND the value of  
238 the column in that location is also between 0 and 4 inclusive.

239 a) Say “Yes”

240 Otherwise

241 a) Say “No”

242

243

---

244 **get**

245

246 When asked to **get**, you will be given a Location

247

248 1) If there is nothing in that cell in your current, private table

249 a) Say “null”

250 Otherwise:

251 a) Say “<The name of the Object at that Location in your private table>.”

252

253

---

254 **put**

255

256 When asked to **put**, you will be given an Actor and Location

257

258 1) Write the name of the Actor you were given into the specified location in your  
259 current, private table. Erase any name that was there before this operation.

260 2) Say, “<name of actor> has been placed in location <given location>.”

261

262

263

264

265

266

267

268 **You are a Grid.**

269

270 **remove**

271

272 When asked to **remove**, you will be given a Location

- 273 1) If there is no Actor at the given Location, then **Say** “There is no Actor to remove,  
274 done removing”.
- 275 2) Erase the name of the Actor you were given into the specified location in your  
276 current, private table, remembering the name erased for just a moment.
- 277 3) Say, “<name of actor erased> has been removed from location <given location>.”

278

279

280

281 **getOccupiedLocations**

282

283 When asked to **getOccupiedLocations**

284

- 285 1) On your Post-it note, write down all of the locations (coordinate pairs, not names)  
286 that are not null according to your current, private table.
- 287 2) Give that Post-it note to the person who made the request, *while Saying*, “Here is  
288 the list of occupied locations.”

289

## 289 You are a Grid.

290

---

### 291 **getNeighbors**

292

293 When asked to **getNeighbors**, you will be given a Location

294

295 1) Narrate, “I will begin by getting a list of *occupied* adjacent locations by calling

296 my helper method **getOccupiedAdjacentLocations**.”

297 2) Do **getOccupiedAdjacentLocations** – this method is located below.

298 3) Take a blank piece of paper (call it the Actors List sheet.)

299 4) Narrate, “I will now get the name of the actor associated with each location.”

300 5) On your new Actors List sheet, write down the *name* of each actor that is

301 associated with a location on the sheet from Step 2 as indicated by your private

302 table.

303 6) Crumple up the sheet from Step 2.

304 7) Narrate, “I have now constructed the list of neighbors.” and Hold the sheet from

305 Step 5 up for the audience to see.

306 8) If the request for this list came from someone else, give that person the list.

307

308

---

### 309 **getOccupiedAdjacentLocations**

310

311 When asked to **getOccupiedAdjacentLocations**, you will be given a Location

312

313 1) Narrate, “I will begin by getting a list of *valid* adjacent locations by calling my

314 helper method **getValidAdjacentLocations**”

315 2) Do **getValidAdjacentLocations** (below).

316 3) Take a blank Location List sheet.

317 4) Narrate, “I will now copy over only those locations that are occupied.”

318 5) On your new Location List sheet, write down all of the locations (coordinate

319 pairs, not names) from the valid locations sheet that are currently occupied

320 according to your private table.

321 6) Crumple up the sheet from Step 2 (the first sheet that you made)

322 7) Narrate, “I have now constructed the list of occupied adjacent locations.” and

323 Hold the sheet from Step 5 up for the audience to see.

324 8) If the request for this list came from someone else, give that person the list. I am

325 done!

326

327

327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362

## You are a Grid.

---

### **getValidAdjacentLocations**

When asked to **getValidAdjacentLocations**, you will be given a Location

- 1) Take a blank Location List sheet.
  - 2) On that sheet, write down all of the locations (coordinate pairs, not names) from your table that are adjacent to the given location STARTING with location that is NORTH of the current location and working CLOCKWISE. DO NOT include any locations that are outside your table due to coordinate values being too low or too high.
  - 3) Narrate, “I have now constructed the list of valid adjacent locations.” and Hold it up for the audience to see.
  - 4) If the request for this list came from someone else, give that person the list.
- 

### **getEmptyAdjacentLocations**

When asked to **getEmptyAdjacentLocations**, you will be given a Location

- 1) Narrate, “I will begin by getting a list of *valid* adjacent locations by calling my helper method **getValidAdjacentLocations()**”
- 2) Do **getValidAdjacentLocations()** – this is located above
- 3) Take a blank Location List sheet.
- 4) Narrate, “I will now copy over only those locations that are empty.”
- 5) On your new Location List sheet, write down all of the locations (coordinate pairs, not names) from the valid locations sheet that are currently empty according to your private table.
- 6) Crumple up the sheet from Step 2.
- 7) Narrate, “I have now constructed the list of empty adjacent locations.” and Hold the sheet from Step 5 up for the audience to see.
- 8) If the request for this list came from someone else, give that person the list.

Location List sheet	
(____,____)	(____,____)
(____,____)	(____,____)
(____,____)	(____,____)
(____,____)	(____,____)
(____,____)	(____,____)

362 **Bug** \_\_\_\_\_ (*your name here*)

363

364

365 **Private information**

366

367

368                    yourGrid: \_\_\_\_\_

369

370                    yourDirection: \_\_\_\_\_

371

372                    yourColor: \_\_\_\_\_

373

374                    yourLocation: ( \_\_\_\_\_ , \_\_\_\_\_ )

375

376

376 **You are a Bug. (“of type Actor”)**

377

378

---

379 **Constructing Yourself**

380

381 When you are **constructed** you will be given:

382

- Your **Color**

383

384 1) Remember your color by recording it in the appropriate places on your **Private**  
385 **Data** sheet. (Note: if you were not given your color, then assume that your color  
386 is “Red” and your direction is “North”.) Leave other data blank for now.

387 2) Say “Done constructing Bug <your name>.”

388

389

---

390 **Are you a \_\_\_\_\_?**

391

392 When asked if you are a **Bug** or an **Actor**:

393

- 1) Say “true”.

394

When asked if you are anything else (i.e., anything other than a **Bug** or an **Actor**):

395

- 1) Say “false”.

396

397

---

398 **act**

399

400 When asked to **act**:

401

402 1) Narrate “Hmm... I must determine if I can move.”

403

2) Ask yourself if you **canMove** (**this script is on the next page**).

404

3) If the answer was that you can move

405

a) then **move** (**this script is on the next page**)

406

Otherwise

407

a) **turn** (**skip two pages to find this script**)

408

4) Say “<Your name> is done acting.”

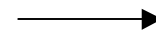
409

410

## You are a Bug. (“of type Actor”)

### canMove

When asked if you **canMove**

- 1) Say “My location is <yourlocation> (row, column)
- 2) Say “My direction is <your direction>
- 3) Say “Therefore, my next location (row, column) would be <the result of moving forward>”
- 4) Name this new Location <nextLocation>. 
- 5) Ask your grid if <nextLocation> **isAValidLocation**
- 6) If the grid says, “No”, then say, “I cannot move” (and return to your **act** script on the previous page.)
- 7) Ask your grid to **get** the name of the object (if not null) at location <nextLocation>.
- 8) If the grid says, “Null”, then Say, “I can move” (and return to your act script)
- 9) Otherwise Ask that person, “Are you a Flower?”
- 10) If that person says, “Yes”, then Say, “I can move” (and return to your **act** script on the previous page)
- 11) If that person says, “No”, then Say, “I cannot move” (and return to your **act** script on the previous page)

<b>nextLocation</b> (____,____)
------------------------------------

**Return to act method script, line 3**

### move

When asked to **move**:

<b>currentLocation</b> (____,____)	<b>nextLocation</b> (____,____)
---------------------------------------	------------------------------------

- 1) If your grid is null
  - a) Say “Done Moving.” (...and return to directions for **act**)
- Otherwise:
  - a) Make a copy of <yourLocation> and name it <currentLocation> Write this in the box above.
  - b) Determine the Location in front of you as described below:
    - i) Say “My location is <your location>
    - ii) Say “My direction is <your direction>
    - iii) Say “Therefore, my next location (row, column) would be <the result of moving forward>”
    - iv) Name this next Location <nextLocation>.
  - c) Ask your Grid if <nextLocation> **isAValidLocation**.
  - d) If the Grid replies positively
    - i) **moveTo** the location <nextLocation> (script on next page)
  - Otherwise
    - i) **removeYourself** from the grid.
  - e) Pick a person not currently involved in the role play and tell them to construct themselves as a Flower with color <yourColor>.
  - f) Tell that person to **putSelfInGrid** into <your grid> at location <currentLocation>.
  - g) Narrate “I am done moving.” (return to **act** script above)

<b>Why are we asking the Grid “isAValidLocation()” again? Didn’t we just do that?</b>
---

<b>How does ActorWorld know that a new Flower is now in the Grid?</b>
---

455 **You are a Bug. (“of type Actor”)**

456

457 **moveTo**

458

459

<b>nextLocation</b> (____,____)
------------------------------------

460 When asked to **move**, you will be given a *<nextLocation>*. (If not, complain.)

461

462 1) Tell your grid to **remove** the object at *<currentLocation>*. That object may stand

463 up and step out of the grid to simulate being removed from the grid.

464 2) Tell the object at *<nextLocation>* (if any) to **removeYourself** (see script below).

465 This object may also stand up and step out of the grid to simulate being removed

466 from the grid.

467 3) Set *<your location>* to *<nextLocation>*, remembering to write it on your private

468 data sheet. I am setting my location to *<nextLocation>*.

469 4) Tell your grid to **put** *<your name>* at *<your location>*.

470

471 **Return to move method script, line 1di**

472

473 **turn**

474 When asked to **turn**:

475 1) Narrate “My current direction is *<your current direction>*.”

476 2) Narrate “The direction 45 degrees clockwise from that is *<the appropriate*

477 *direction>*.”

478 3) Set your direction to that direction, remembering to write it on your private data

479 sheet.

480 4) Say, “I am done turning.”

481

482 **Return to the method that called turn()**

483

484 **putSelfInGrid**

485 When asked to **putSelfInGrid**, you will be given a grid and location. Write these on

486 your private data sheet.

487 1) Narrate “I need to be placed into the grid”

488 2) Ask *<your grid>* to “**put**” *<your name>* at location *<the location you were given>*

489 3) Say “Done putting myself into the grid.”

490

491 **removeYourself**

492 When asked to **removeYourself**

493 1) Narrate “I need to get out of the grid, but I can’t do this alone.”

494 2) Ask your grid to **remove** *<your name>*

495 3) Set your grid to null

496 4) Set your location to null. This object may also stand up and step out of the grid to

497 simulate being removed from the grid.

498 5) Say “Done removing myself from the grid.”

499

499 **Rock** \_\_\_\_\_ (*your name here*)

500

501

502 **Private information**

503

504 **your Grid:** \_\_\_\_\_

505

506 **your Color:** \_\_\_\_\_

507

508 **your Location:** ( \_\_\_\_\_ , \_\_\_\_\_ )

509

509 **You are a Rock. (“of type Actor”)**

510

511

---

512 **Constructing Yourself**

513

514 When you are **constructed** you might be given your color. If not given your color,  
515 assume that it is black.

516

517 • **Your Color**

518

519 1) Remember your color by recording it in the appropriate places on your **Private**  
520 **Data** sheet. (Note: if you were not given your color, then assume that your color  
521 is “Black” and direction “North”). Leave other data blank for now.

522 2) Say “Done constructing Rock <your name>.”

523

524

---

525 **Are you a \_\_\_\_\_?**

526

527 When asked if you are a **Rock** or an **Actor**:

528 1) Say “true”.

529 When asked if you are anything else (i.e., anything other than a **Rock** or an  
530 **Actor**):

531 1) Say “false”.

532

---

533 **act**

534

535 When asked to **act**

536

537 1) Say “<Your name> is done acting.”

538

539

539 **You are a Rock. (“of type Actor”)**

540

541 **putSelfInGrid**

542

543 When asked to **putSelfInGrid**, you will be given a grid and location. Write these on  
544 your private data sheet.

545

546 1) Narrate “I need to be placed into the grid”

547 2) Ask your grid to “**put**” <your name> at location <the location you were given>

548 3) Say “Done putting myself into the grid.”

549

550

551 **removeYourself**

552

553 When asked to **removeYourself**

554

555 1) Narrate “I need to get out of the grid, but I can’t do this alone.”

556 2) Ask your grid to **remove** <your name>

557 3) Set your grid to null

558 4) Set your location to null

559 5) Say “Done removing myself from the grid.”

560

560 **Flower** \_\_\_\_\_ (*your name here*)

561

562

563 **Private information**

564

**your Grid:** \_\_\_\_\_

565

566

**your Color:** \_\_\_\_\_

567

568

**your Location:** ( \_\_\_\_\_ , \_\_\_\_\_ )

569

570

571

571 **You are a Flower. (“of type Actor”)**

572

573 **Constructing Yourself**

574

575 When you are **constructed** you will be given:

576

- Your **Color**

577

578 1) Remember your color by recording it in the appropriate places on your **Private**  
 579 **Data** sheet. (Note: if you were not given your color, then assume that your color  
 580 is “Pink” and direction “North”). Leave other data blank for now.

581

2) Say “Done constructing Flower <your name>.”

582

583

584 **Are you a \_\_\_\_\_?**

585

586 When asked if you are a **Flower** or an **Actor**:

587

- 1) Say “true”.

588

589 When asked if you are anything else (i.e., anything other than a **Flower** or an  
**Actor**):

590

- 1) Say “false”.

591

592 **act**

593

594 When asked to **act**:

595

- 1) Narrate “Hmm... I must darken myself.”
- 2) Using the table below, set your color to the next darkest color. If you are already at “Black” just stay there.
- 3) Narrate “My new color is <your color>.”
- 4) Say “<Your name> is done acting.”

601

602

603

Darkening table: Assumes you start at “red”

Pink
Red
Deep red
Slightly dark red
Pretty dark red
Dark red
Very dark red
Reddish black
Black with a hint of red
Essentially black
Black

604

604 **You are a Flower. (“of type Actor”)**

605

606 **putSelfInGrid**

607

608 When asked to **putSelfInGrid**, you will be given a grid and location. Write these on  
609 your private data sheet.

610

- 611 1) Narrate “I need to be placed into the grid”
- 612 2) Ask your grid to “**put**” <your name> at location <the location you were given>
- 613 3) Say “Done putting myself into the grid.”

614

615

616

617 **removeYourself**

618

619 When asked to **removeYourself**

620

- 621 1) Narrate “I need to get out of the grid, but I can’t do this alone.”
- 622 2) Ask your grid to **remove** <your name> at location <your location>.
- 623 3) Set your grid to null
- 624 4) Set your location to null
- 625 5) Say “Done removing myself from the grid.”

626

## 626 You are the RolePlayCriticRunner

627

628 **Run**

629

630 When you the role play is set to begin

- 631 1) Construct an ActorWorld by picking a pre-selected person in the room and
- 632 saying, <person name>, “Construct yourself as an ActorWorld”
- 633 2) Construct a Critter by picking a pre-selected person in the room and saying,
- 634 <person name>, “Construct yourself as an Critter”
- 635 3) Ask your ActorWorld to **add** the Critter <person name> at position (0,0)
- 636 4) Construct a Rock by picking a pre-selected person in the room and saying,
- 637 <person name>, “Construct yourself as an Rock”
- 638 5) Ask your ActorWorld to **add** your Rock <person name> at position (1,0)
- 639 6) Construct a Bug by picking a pre-selected person in the room and saying,
- 640 <person name>, “Construct yourself as an Bug; your color is Blue”
- 641 7) Ask your ActorWorld to **add** your Bug <person name> at position (0,1)
- 642 8) Construct a Bug by picking a pre-selected person in the room and saying,
- 643 <person name>, “Construct yourself as an Bug; your color is Yellow”
- 644 9) Ask your ActorWorld to **add** your Bug <person name> at position (1,2)
- 645 10) Construct a Critter by picking a pre-selected person in the room and saying,
- 646 <person name>, “Construct yourself as an Critter”
- 647 11) Ask your ActorWorld to **add** your Critter <person name> at position (2,2)
- 648 12) Construct a Bug by picking a pre-selected person in the room and saying,
- 649 <person name>, “Construct yourself as a Bug”
- 650 13) Ask your ActorWorld to **add** your Bug <person name> at position (3,2)
- 651 14) Have a **commercialInterruption**
- 652 15) Ask your ActorWorld to **step**.
- 653 16) Have a **commercialInterruption**
- 654 17) Ask your ActorWorld to **step**.
- 655 16) Have a **commercialInterruption**
- 656

## You are the RolePlayCriticRunner

---

656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674

### commercialInterruption

To process a **commercialInterruption**

- 1) Narrate “We now interrupt this role play for a commercial interruption. The sponsors hope that you will pay attention to the display during this break.”
- 2) For each Actor (e.g. Bug, Flower, Rock, Critter – NOT ActorWorld or Grid) who is currently involved in the role play, do the following:
  - a) Ask the actor if their Grid is null
    - i) If the actor says, “yes”, then skip to the next actor.
  - b) Ask the actor for his/her location.
  - c) Ask the actor for his/her color.
  - d) Ask the actor for his/her direction.
  - e) Update whatever display your instructor has provided (e.g. blackboard, whiteboard, poster, etc.) to reflect this actor’s status in the world.
- 3) Narrate “No more actors. We now resume our regularly scheduled program.”  
(Feel free to emphasize the last word...)

675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687

**Critter** \_\_\_\_\_ (*your name here*)

**Private information**

**your Grid:** \_\_\_\_\_

**your Direction:** \_\_\_\_\_

**your Color:** \_\_\_\_\_

**your Location:** ( \_\_\_\_\_ , \_\_\_\_\_ )

687 **You are a Critter. (“of type Actor”)**

688

689

---

690 **Constructing Yourself**

691

692 When **constructed**, assume that your color is “Blue” and your direction is “North.”

693

694 1) Remember your color and direction by recording it in the appropriate places on  
695 your **Private Data** sheet.

696 2) Say “Done constructing Critter <your name>.”

697

698

---

699 **Are you a \_\_\_\_\_?**

700

701 When asked if you are a **Critter** or an **Actor**:

702 1) Say “true”.

703 When asked if you are anything else (i.e., anything other than a **Critter** or an  
704 **Actor**):

705 1) Say “false”.

706 2)

707

---

708 **putSelfInGrid**

709

710 When asked to **putSelfInGrid**, you will be given a grid and location. Write these on  
711 your private data sheet.

712

713 1) Narrate “I need to be placed into the grid”

714 2) Ask your grid to “**put**” <your name> at location <the location you were given>

715 3) Say “Done putting myself into the grid.”

716

---

717 **removeYourself**

718

719 When asked to **removeYourself**

720

721 1) Narrate “I need to get out of the grid, but I can’t do this alone.”

722 2) Ask your grid to **remove** <your name>

723 3) Set your grid to null

724 4) Set your location to null

725 5) Say “Done removing myself from the grid.”

726

726 **You are a Critter. (“of type Actor”)**

727

728 **act**

729

730 When asked to **act**:

731

732 1) Narrate “Ahh, to act... First I will begin by getting a list of the objects with  
733 whom I might interact.”

734 2) Narrate “Ahh, I must ask myself to **getActors** by calling my helper method  
735 **getActors.**” (Jump to that script on the next page)

736 3) Narrate “Behold my list of actors!! Now to continue action. Since I have the list  
737 of actors, I must now process it.”

738 4) Narrate “Ahh, I must ask myself to **processActors** by calling my helper method  
739 **processActors.** (Jump to that script on the next page)

740 5) Narrate “Now that I am done processing everyone, I can think about moving. I  
741 will start by getting list of possible locations.

742 6) Narrate “Ahh, I must ask myself to **getMoveLocations** by calling my helper  
743 method **getMoveLocations.** (Jump to that script on the next page)

744 7) Narrate “Ahh, the list of locations...now I must choose one...”

745 8) Narrate “Ahh, I must ask myself to **selectMoveLocation** by calling my helper  
746 method **selectMoveLocation.** (Jump to that script on the next page)

747 9) Narrate “I have chosen *<the chosen location>* and will now make my move”

748 10) Narrate “Ahh, I must ask myself to **makeMove** to *<the chosen location>* by  
749 calling my helper method **makeMove.** (Jump to that script on the next page)

750 11) Narrate “At long, last, I am done acting.”

751

751 **You are a Critter. (“of type Actor”)**

752

753 **getActors**

754

755 When asked to **getActors**

756

757 1) Narrate “Hmmm... With whom do I act? With my neighbors, of course. I’d  
758 better ask the grid who they are.”

759 2) Ask <your grid> to **getNeighbors** of <your location>

760 3) Wave the list given to you by the grid and narrate “Behold my list of actors!”

761

762 [Return to Step 3 of **act**.]

763

764

765 **processActors**

766

767 When asked to **processActors**, you should already have a list of actors...

768

769 1) Narrate “Time to eat... Who is near by that is edible?”

770 2) For each actor in your list

771 a) Ask <that actor> if it is a Rock.

772 i) If it says, “Yes”, dramatically cross it off the list and move to the next  
773 actor in the list.

774 b) Ask <that actor> if it is a Critter.

775 i) If it says, “Yes”, dramatically cross it off the list and move to the next  
776 actor in the list.

777 c) Ask <that actor> to **removeYourself**

778 d) Dramatically cross <that actor> off the list and move to the next actor in the  
779 list.

780 3) Narrate “That’s the whole list!”

781

782 [Return to Step 5 of **act**.]

783

## 783 You are a Critter. (“of type Actor”)

---

### 784 **getMoveLocations**

785  
786  
787 When asked to **getMoveLocations**

- 788
- 789 1) Narrate “Hmmm... Where can I move? To any adjacent empty space. I’d better
  - 790 ask the grid which ones those are.”
  - 791 2) Ask *<your grid>* to **getEmptyAdjacentLocations** of *<your location>*
  - 792 3) Wave the list given to you by the grid and narrate “Behold my list of possible
  - 793 destinations!”

794  
795 [Return to Step 7 of **act.**]

---

### 796 **selectMoveLocation**

797  
798  
799  
800 When asked to **selectMoveLocation**, you should already have a list of locations...

- 801
- 802 1) Narrate “Time to move...There are *<number of items in your list>* possibilities.”
  - 803 2) Ask *<The Random Number Generator>* for a number up from “0” to the *<number*
  - 804 *of items in your list -1 >* (to be consistent with array notation...)
  - 805 3) Narrate “Let’s see. The *<number you were given>*th item in my list is *<that*
  - 806 *location>*. I’ll move there!”

807  
808 [Return to Step 9 of **act.**]

---

809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827

Why is there a separate makeMove method? Why not just call moveTo?

828 **makeMove**

829

830 When asked to **makeMove**, you will be given a location.

831

832 1) Narrate “Time to move...my choice is to move to *<that location>*.”

833 2) Narrate “Actually, I do this the same way any actor does...”

834 3) Ask yourself to **moveTo** *<that location>*. (see script below)

835

836 [Return to Step 11 of **act**.]

837

838

---

839 **moveTo**

840

841 When asked to **moveTo**, you will be given a **newLocation**. (If not, complain.)

842

843 1) Tell *<your grid>* to **remove** the object at *<your current location>*.

844 2) Tell the object at *<nextLocation>* (if any) to **removeYourself** (see script below).

845 This object may also stand up and step out of the grid to simulate being removed  
846 from the grid.

847 3) Set *<your location>* to *<newLocation>*, remembering to write it on your private  
848 data sheet.

849 4) Tell your grid to **put** *<your name>* at *<your location>*

850 g) Say “I have completed my move and will return to my **makeMove** script.”