```java
/**
 * TourActivity.java
 * Brock Butler
 * Main activity for the tour portion of Brock Butler.
 * Created by Taras Mychaskiw 2013-02-20
 * Copyright (c) 2013 Sea Addicts. All rights reserveR.drawable.
 */
package edu.seaaddicts.brockbutler.tour;

import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.ImageButton;
import android.widget.RelativeLayout;
import android.widget.Toast;
import edu.seaaddicts.brockbutler.R;

public class TourActivity extends Activity {
  private TourNode[] nodes;
  private final int numImages = R.drawable.j328 - R.drawable._a301b + 1;
  private TourInfo info;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_tour);

    RelativeLayout rl = (RelativeLayout)findViewById(R.id.screen);
    ImageButton[] buttons = new ImageButton[5];
    buttons[0] = (ImageButton)findViewById(R.id.outerleft);
    buttons[1] = (ImageButton)findViewById(R.id.innerleft);
    buttons[2] = (ImageButton)findViewById(R.id.center);
    buttons[3] = (ImageButton)findViewById(R.id.innerright);
    buttons[4] = (ImageButton)findViewById(R.id.outerright);
    info = new TourInfo(rl,buttons,getApplicationContext());
    initNodes();
    nodes[idx(R.drawable._a301f)].paint(info);
  }

  @Override
  public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.activity_tour, menu);
    return true;
  }

  /**
   * Pops the previous TourNode from the stack and goes to that node.
   */
  public void goBack(MenuItem item){
    if (!info.history.empty())
      info.history.pop().paint(info);
  }

  /**
   * Goes to the node in the tour which is logically turning around.
   */
  public void turnAround(MenuItem item){
    if (info.current.canTurnAround()){
      info.history.push(info.current);
      info.current.turnAroundNode.paint(info);
    }
    else
      Toast.makeText(info.context,"No data available",Toast.LENGTH_SHORT).show();
  }

  /**
   * Teleports the user to the specified block.
   */
  public void teleport(MenuItem item){
    info.history.push(info.current);
    switch (item.getItemId()){
    case R.id.teleport_a_block: nodes[idx(R.drawable._a301f)].paint(info); return;
    case R.id.teleport_b_block: nodes[idx(R.drawable._b301f)].paint(info); return;
    case R.id.teleport_c_block: nodes[idx(R.drawable._c301f)].paint(info); return;
    case R.id.teleport_d_block: nodes[idx(R.drawable._d301b)].paint(info); return;
    case R.id.teleport_e_block: nodes[idx(R.drawable._e301f)].paint(info); return;
    case R.id.teleport_f_block: nodes[idx(R.drawable._f301f)].paint(info); return;
    case R.id.teleport_g_block: nodes[idx(R.drawable._h338f)].paint(info); return;
    case R.id.teleport_h_block: nodes[idx(R.drawable._f301f)].paint(info); return;
    case R.id.teleport_j_block: nodes[idx(R.drawable._j301f)].paint(info); return;
    }
  }

  /**
   * Ends the tour and destroys the Activity
   */
  public void endTour(MenuItem item){
    super.onBackPressed();
  }

  /**
   * Overrides the back button to pop previous TourNode from the stack
   * and goes to that node.
   */
  @Override
  public void onBackPressed(){
    goBack(null);
  }

  /**
   * @param r - image resource value
   * @return index in `nodes` of TourNode which shows the image `r`
   */
  private int idx(int r){
    return(r - R.drawable._a301b);
  }

  /**
   * Adds a new TourRoom to nodes.
   * @param r - the image resource value
   * @param roomNumber - the title of the room, such as A323
   */
  private void room(int r, String roomNumber){
    nodes[idx(r)] = new TourRoom(r,roomNumber);
  }

  /**
   * Adds a new TourHall to nodes.
   * @param r  - the image resource value
   * @param ol - image resource value of outer left button
   * @param ul - resource of inner left
   * @param c  - resource of center
   * @param ur - resource of inner right
   * @param or - resource of outer right
   * @param ta - resource of turn around node
   */
  private void hall(int r, int ll, int ul, int c, int ur, int lr, int ta){
    if (ta != -1)
      nodes[idx(r)] = new TourHall(r,
        (ll == -1)? null : nodes[idx(ll)],
        (ul == -1)? null : nodes[idx(ul)],
        (c == -1) ? null : nodes[idx(c)],
        (ur == -1)? null : nodes[idx(ur)],
        (lr == -1)? null : nodes[idx(lr)],
        /*(ta == -1)?*/nodes[idx(ta)]);
    else
      nodes[idx(r)] = new TourHall(r,
        (ll == -1)? null : nodes[idx(ll)],
        (ul == -1)? null : nodes[idx(ul)],
        (c == -1) ? null : nodes[idx(c)],
```

```java
          (ur == -1)? null : nodes[idx(ur)],
          (lr == -1)? null : nodes[idx(lr)]);
    }
    private void hall(int r, int c){ hall(r,-1,-1,c,-1,-1,-1); }
    private void hall(int r, int c, int ta){ hall(r,-1,-1,c,-1,-1,ta); }
    private void hall (int r, int ll, int ul, int c, int ur, int lr){ hall(r,ll,ul,c,ur,lr,-1); }

    /**
     * Giant method to link all of the images together. The fun stuff. Right here.
     */
    private void initNodes(){
      nodes = new TourNode[numImages];

      /** A BLOCK ROOMS **/
      room(R.drawable.a323,"A323");

      /** B BLOCK ROOMS **/
      room(R.drawable.b303a,"B303A");
      room(R.drawable.b303b,"B303B");
      hall(R.drawable.b303,-1,-1,R.drawable.b303b,R.drawable.b303a,-1);

      /** C BLOCK ROOMS **/
      room(R.drawable.c300,"C300");
      room(R.drawable.c303,"C303");
      room(R.drawable.c304,"C304");
      room(R.drawable.c306_1,"C306");
      room(R.drawable.c306_2,"C306");
      hall(R.drawable.c306_3,R.drawable.c306_2,-1,-1,-1,R.drawable.c306_1);
      room(R.drawable.c308,"C308");
      room(R.drawable.c310,"C310");

      /** D BLOCK ROOMS **/
      room(R.drawable.d300,"D300");
      room(R.drawable.d301,"D301");
      room(R.drawable.d303,"D303");
      room(R.drawable.d304,"D304");
      room(R.drawable.d308,"D308");
      room(R.drawable.d309_1,"D309");
      room(R.drawable.d309_2,"D309");
      room(R.drawable.d310,"D310");
      room(R.drawable.d314,"D314");
      room(R.drawable.d315,"D315");
      room(R.drawable.d316,"D316");
      room(R.drawable.d317,"D317");
      room(R.drawable.d318,"D318");
      room(R.drawable.d319,"D319");
      room(R.drawable.d350l,"D350L");

      /** E BLOCK ROOMS **/
      room(R.drawable.e302,"E302");
      room(R.drawable.e303,"E303");
      room(R.drawable.e304,"E304");

      /** F BLOCK ROOMS **/
      /* N/A */
      /** G BLOCK ROOMS **/
      room(R.drawable.g301_2,"G301");
      room(R.drawable.g301_3,"G301");
      hall(R.drawable.g301_1,R.drawable.g301_2,-1,-1,-1,R.drawable.g301_3);
      room(R.drawable.g305,"G305");

      /** H BLOCK ROOMS **/
      room(R.drawable.h300,"H300");
      room(R.drawable.h302,"H302");
      room(R.drawable.h303,"H303");
      room(R.drawable.h304a,"H304A");
      hall(R.drawable.h304,-1,-1,-1,-1,R.drawable.h304a);
      hall(R.drawable.h306,/*R.drawable.h309a6*/-1,-1,-1,/*R.drawable.h306a6*/-1,-1);
      hall(R.drawable.h309,-1,/*R.drawable.h306a9*/-1,-1,-1,/*R.drawable.h309a9*/-1);
      hall(R.drawable.h306a6,-1,-1,-1,R.drawable.h309,-1);
```

```
        hall(R.drawable.h306a9,-1,R.drawable.h306,-1,-1,-1,R.drawable.h306a6);
        hall(R.drawable.h309a6,R.drawable.h309,-1,-1,-1,-1);
        hall(R.drawable.h309a9,R.drawable.h306,-1,-1,-1,-1,R.drawable.h309a6);
        nodes[idx(R.drawable.h306)].setOuterLeftNode(nodes[idx(R.drawable.h309a6)]);
        nodes[idx(R.drawable.h306)].setInnerRightNode(nodes[idx(R.drawable.h306a6)]);
        nodes[idx(R.drawable.h309)].setInnerLeftNode(nodes[idx(R.drawable.h309a9)]);
        nodes[idx(R.drawable.h309)].setOuterRightNode(nodes[idx(R.drawable.h306a9)]);
        room(R.drawable.h310,"H310");
        room(R.drawable.h313,"H313");
        room(R.drawable.h315,"H315");
        room(R.drawable.h316,"H316");
        room(R.drawable.h317,"H317");
        room(R.drawable.h318a,"H318A");
        hall(R.drawable.h318,-1,-1,-1,R.drawable.h318a,-1);
        room(R.drawable.h320,"H320");
        room(R.drawable.h321,"H321");
        room(R.drawable.h322,"H322");
        room(R.drawable.h323,"H323");
        room(R.drawable.h324,"H324");

        /** J BLOCK ROOMS **/
        room(R.drawable.j301,"J301");
        room(R.drawable.j310,"J310");
        room(R.drawable.j327,"J327");
        room(R.drawable.j328,"J328");

        /** J BLOCK FORWARD PASS **/
        hall(R.drawable._j315f,-1,-1,-1,-1,R.drawable.j301);
        hall(R.drawable._j314f,R.drawable._j315f);
        hall(R.drawable._j313f,R.drawable._j314f);
        hall(R.drawable._j312f,-1,-1,-1,R.drawable._j313f,-1);
        hall(R.drawable._j311f,R.drawable._j312f);
        hall(R.drawable._j310f,R.drawable._j311f);
        hall(R.drawable._j309f,R.drawable._j310f);
        hall(R.drawable._j308f,R.drawable._j309f);
        hall(R.drawable._j307f,-1);
        hall(R.drawable._j306f,R.drawable._j307f,-1,-1,-1,R.drawable._j308f);
        hall(R.drawable._j305f,R.drawable._j306f);
        hall(R.drawable._j304f,-1,R.drawable.j327,R.drawable._j305f,R.drawable.j328,-1);
        hall(R.drawable._j303f,R.drawable._j304f);
        hall(R.drawable._j302f,-1,R.drawable.j310,-1,R.drawable.j301,-1);
        hall(R.drawable._j301f,-1,-1,R.drawable._j303f,R.drawable._j302f,-1);

        /** D BLOCK FORWARD PASS **/
        hall(R.drawable._d343f,/*R.drawable._d304b*/-1,-1,-1,/*R.drawable._j312b*/-1,-1);
        hall(R.drawable._d342f,R.drawable._d343f);
        hall(R.drawable._d341f,-1,R.drawable.g310,R.drawable._j301f,R.drawable._d342f,-1);
        hall(R.drawable._d340f,-1,-1,R.drawable._d341f,-1,/*R.drawable._d307b*/-1);
        hall(R.drawable._d339f,-1,-1,R.drawable._d340f,/*R.drawable._d307b*/-1,R.drawable.d309_2);
        hall(R.drawable._d338f,R.drawable._d339f);
        hall(R.drawable._d337f,-1,-1,R.drawable._d338f,R.drawable.d309_1,-1);
        hall(R.drawable._d336f,R.drawable._d337f);
        hall(R.drawable._d335f,/*R.drawable._d306b*/-1,-1,-1,-1,R.drawable._d341f);
        hall(R.drawable._d334f,-1,-1,R.drawable._d335f,-1,R.drawable.d310);
        hall(R.drawable._d333f,-1,-1,R.drawable._d334f,-1,R.drawable.d314);
        hall(R.drawable._d332f,R.drawable._d333f);
        hall(R.drawable._d331f,-1,-1,R.drawable._d332f,-1,R.drawable.d315);
        hall(R.drawable._d330f,-1,-1,R.drawable._d331f,-1,R.drawable.d316);
        hall(R.drawable._d329f,-1,-1,R.drawable._d330f,-1,R.drawable.d317);
        hall(R.drawable._d328f,-1,R.drawable._d329f,R.drawable.d318,-1,-1);
        hall(R.drawable._d327f,R.drawable._d328f);
        hall(R.drawable._d326f,R.drawable.d319,-1,R.drawable._d327f,-1,-1);
        hall(R.drawable._d325f,R.drawable._d326f);
        hall(R.drawable._d324f,R.drawable._d325f);
        hall(R.drawable._d323f,R.drawable._d324f);
        hall(R.drawable._d322f,R.drawable._d323f);
        hall(R.drawable._d321f,R.drawable._d322f);
        hall(R.drawable._d320f,R.drawable._d321f);
        hall(R.drawable._d319f,R.drawable._d320f);
        hall(R.drawable._d318f,-1,-1,R.drawable._d319f,/*R.drawable._d326b*/-1,-1);
        hall(R.drawable._d317f,-1,R.drawable._d336f,R.drawable.d308,R.drawable._d318f,-1);
        hall(R.drawable._d316f,R.drawable.d304,-1,R.drawable._d317f,-1,-1);
        hall(R.drawable._d315f,R.drawable.d303,-1,R.drawable._d316f,-1,-1);
        hall(R.drawable._d314f,R.drawable._d315f);
        hall(R.drawable._d313f,R.drawable.d301,-1,R.drawable._d314f,-1,-1);
        hall(R.drawable._d312f,/*R.drawable._d319b*/-1,-1,-1,-1,R.drawable._d319f);
        hall(R.drawable._d311f,R.drawable._d312f);
        hall(R.drawable._d310f,-1,R.drawable._d311f,-1,-1,-1);
        hall(R.drawable._d309f,R.drawable._d310f);
        hall(R.drawable._d308f,R.drawable._d309f);
        hall(R.drawable._d307f,R.drawable._d308f);
        hall(R.drawable._d306f,R.drawable._d307f);
        hall(R.drawable._d305f,R.drawable._d306f);
        hall(R.drawable._d304f,-1,R.drawable.d350l,-1,-1,-1);
        hall(R.drawable._d303f,-1,R.drawable._d305f,-1,R.drawable._d304f,-1);
        hall(R.drawable._d302f,R.drawable.d300,-1,R.drawable._d313f,-1,R.drawable._d303f);
        hall(R.drawable._d301f,R.drawable._d302f);

        /** C BLOCK FORWARD PASS **/
        hall(R.drawable._c312f,-1,-1,-1,R.drawable._d301f,-1);
        hall(R.drawable._c311f,-1,R.drawable.c310,R.drawable._c312f,-1,-1);
        hall(R.drawable._c310f,R.drawable._c311f);
        hall(R.drawable._c309f,R.drawable.c306_3,-1,R.drawable._c310f,-1,-1);
        hall(R.drawable._c308f,R.drawable._c309f);
        hall(R.drawable._c307f,-1,R.drawable._c308f,-1,-1,-1);
        hall(R.drawable._c306f,-1,-1,/*R.drawable._c306b*/-1,R.drawable._c307f,-1);
        hall(R.drawable._c305f,R.drawable.c304,-1,R.drawable._c306f,-1,-1);
        hall(R.drawable._c304f,R.drawable.c303,-1,R.drawable._c305f,-1,-1);
        hall(R.drawable._c303f,R.drawable._c304f);
        hall(R.drawable._c302f,R.drawable._c303f);
        hall(R.drawable._c301f,-1,-1,-1,R.drawable._c302f,-1);
        hall(R.drawable._c316f,-1,R.drawable._c306f,/*R.drawable._c310b*/-1,-1,-1);
        hall(R.drawable._c315f,R.drawable._c316f);
        hall(R.drawable._c314f,R.drawable.c308,-1,R.drawable._c315f,-1,-1);
        hall(R.drawable._c313f,-1,-1,-1,R.drawable._c314f,-1);

        /** F BLOCK FORWARD PASS **/
        hall(R.drawable._f308f,R.drawable._c313f);
        hall(R.drawable._f307f,R.drawable._f308f);
        hall(R.drawable._f306f,R.drawable._f307f);
        hall(R.drawable._f305f,R.drawable._f306f);
        hall(R.drawable._f304f,R.drawable._f305f);
        hall(R.drawable._f303f,R.drawable._f304f);
        hall(R.drawable._f302f,R.drawable._f303f);
        hall(R.drawable._f301f,-1,R.drawable._f302f,-1,-1,/*R.drawable._h301b*/-1);

        /** G BLOCK FORWARD PASS **/
        hall(R.drawable._g313f,-1);
        hall(R.drawable._g312f,R.drawable._g313f);
        hall(R.drawable._g311f,R.drawable.g305,-1,R.drawable._g312f,-1,-1);
        hall(R.drawable._g310f,R.drawable._g311f);
        hall(R.drawable._g309f,R.drawable._g310f);
        hall(R.drawable._g308f,R.drawable._g309f);
        hall(R.drawable._g307f,R.drawable._g308f);
        hall(R.drawable._g306f,R.drawable._g307f);
        hall(R.drawable._g305f,R.drawable.g301_1,-1,R.drawable._g306f,-1,-1);
        hall(R.drawable._g304f,R.drawable._g305f);
        hall(R.drawable._g303f,R.drawable._g304f);
        hall(R.drawable._g302f,-1,-1,R.drawable._g303f,-1,-1);
        hall(R.drawable._g301f,-1,R.drawable._g302f,-1,-1,-1,R.drawable._f301f);

        /** H BLOCK FORWARD PASS **/
        hall(R.drawable._h338f,-1,R.drawable._g301f,-1,R.drawable._f302f,-1);
        hall(R.drawable._h337f,R.drawable._h338f);
        hall(R.drawable._h336f,R.drawable._h337f);
        hall(R.drawable._h335f,R.drawable._h336f);
        hall(R.drawable._h334f,R.drawable._h335f);
        hall(R.drawable._h333f,-1,/*R.drawable._h303b*/-1,R.drawable._h334f,-1,-1);
        hall(R.drawable._h332f,R.drawable._h333f);
        hall(R.drawable._h331f,R.drawable._h332f);
        hall(R.drawable._h330f,R.drawable._h331f);
        hall(R.drawable._h329f,-1,R.drawable._h334f,/*R.drawable._h303b*/-1,-1,
        /*R.drawable._h322b*/-1);
        hall(R.drawable._h328f,-1,R.drawable.h324,R.drawable._h329f,-1,-1);
        hall(R.drawable._h327f,R.drawable.h323,-1,R.drawable._h328f,-1,-1);
        hall(R.drawable._h326f,R.drawable._h327f);
        hall(R.drawable._h325f,R.drawable._h326f);
        hall(R.drawable._h324f,R.drawable.h320,-1,R.drawable._h325f,-1,-1);
        hall(R.drawable._h323f,-1,-1,R.drawable._h324f,R.drawable.h322,R.drawable.h321);
        hall(R.drawable._h322f,R.drawable.h318,-1,R.drawable._h323f,-1,-1);
        hall(R.drawable._h321f,-1,R.drawable.h316,R.drawable._h322f,-1,R.drawable.h317);
        hall(R.drawable._h320f,R.drawable._h321f);
        hall(R.drawable._h319f,-1,-1,-1,-1,R.drawable._h320f);
        hall(R.drawable._h318f,-1,R.drawable.h315,R.drawable._h319f,-1,-1);
        hall(R.drawable._h317f,R.drawable.h313,-1,R.drawable._h318f,-1,-1);
        hall(R.drawable._h316f,R.drawable._h317f);
        hall(R.drawable._h315f,R.drawable._h316f);
        hall(R.drawable._h314f,-1);
        hall(R.drawable._h313f,R.drawable._h314f);
        hall(R.drawable._h312f,-1,R.drawable.h310,R.drawable._h313f,R.drawable._h315f,-1);
        hall(R.drawable._h311f,-1,-1,-1,R.drawable.h310,R.drawable._h312f);
        hall(R.drawable._h310f,-1,-1,R.drawable._h311f,R.drawable.h309,-1);
        hall(R.drawable._h309f,-1,-1,R.drawable._h310f,-1,R.drawable.h306);
        hall(R.drawable._h308f,-1,-1,R.drawable._h309f,-1,R.drawable.h304);
        hall(R.drawable._h307f,R.drawable.h303,-1,R.drawable._h308f,-1,-1);
        hall(R.drawable._h306f,R.drawable._h307f);
        hall(R.drawable._h305f,-1,-1,R.drawable._h306f,-1,R.drawable.h302);
        hall(R.drawable._h304f,-1,R.drawable.h300,R.drawable._h305f,-1,-1);
        hall(R.drawable._h303f,-1,R.drawable._h304f,-1,R.drawable._h330f,-1);
        hall(R.drawable._h302f,R.drawable._h303f);
        hall(R.drawable._h301f,R.drawable._h302f);

        /** E BLOCK FORWARD PASS **/
        hall(R.drawable._e311f,-1);
        hall(R.drawable._e310f,-1,R.drawable._h301f,-1,R.drawable._e311f,-1);
        hall(R.drawable._e309f,R.drawable._e310f);
        hall(R.drawable._e308f,R.drawable._e309f);
        hall(R.drawable._e307f,-1,-1,R.drawable._e308f,-1,-1);
        hall(R.drawable._e306f,R.drawable.e304,-1,R.drawable._e307f,-1,-1);
        hall(R.drawable._e305f,R.drawable.e303,-1,R.drawable._e306f,-1,-1);
        hall(R.drawable._e304f,R.drawable._e305f);
        hall(R.drawable._e303f,-1,R.drawable.e302,R.drawable._e304f,-1,-1);
        hall(R.drawable._e302f,R.drawable._e303f);
        hall(R.drawable._e301f,-1,-1,-1,R.drawable._e302f,-1);

        /** B BLOCK FORWARD PASS **/
        hall(R.drawable._b314f,R.drawable._c301f);
        hall(R.drawable._b313f,R.drawable._b314f);
        hall(R.drawable._b312f,R.drawable._b314f);
        hall(R.drawable._b311f,-1,-1,-1,R.drawable._b312f,-1);
        hall(R.drawable._b310f,-1,R.drawable._b311f,R.drawable._b313f,-1,-1);
        hall(R.drawable._b309f,R.drawable._b310f);
        hall(R.drawable._b308f,-1,R.drawable._b309f,-1,-1,-1);
        hall(R.drawable._b307f,R.drawable._e301f);
        hall(R.drawable._b306f,R.drawable._b307f);
        hall(R.drawable._b305f,R.drawable._b306f);
        hall(R.drawable._b304f,-1,-1,R.drawable._b305f,R.drawable.b303,-1);
        hall(R.drawable._b303f,R.drawable._b304f);
        hall(R.drawable._b302f,-1,R.drawable._b303f,-1,R.drawable._b308f,-1);
        hall(R.drawable._b301f,-1,-1,-1,R.drawable._b302f,-1);

        /** A BLOCK FORWARD PASS **/
        hall(R.drawable._a309f,-1,R.drawable._b301f,-1,-1,-1);
        hall(R.drawable._a308f,R.drawable._a309f);
        hall(R.drawable._a307f,R.drawable._a308f);
        hall(R.drawable._a306f,R.drawable._a307f);
        hall(R.drawable._a305f,-1,-1,R.drawable._a306f,-1,R.drawable.a323);
        hall(R.drawable._a304f,R.drawable._a305f);
        hall(R.drawable._a303f,R.drawable._a304f);
        hall(R.drawable._a302f,R.drawable._a303f);
        hall(R.drawable._a301f,R.drawable._a302f);

        /** A BLOCK BACK PASS **/
        hall(R.drawable._a307b,-1,R.drawable._a302f);
        hall(R.drawable._a306b,R.drawable._a307b,R.drawable._a303f);
        hall(R.drawable._a305b,R.drawable._a306b,R.drawable._a304f);
        hall(R.drawable._a304b,-1,R.drawable.a323,R.drawable._a305b,-1,-1,R.drawable._a307f);
        hall(R.drawable._a303b,R.drawable._a304b,R.drawable._a308f);
        hall(R.drawable._a302b,-1,-1,-1,R.drawable._a303b,-1,R.drawable._a309f);
        hall(R.drawable._a301b,R.drawable._a302b,R.drawable._b301f);

        /** B BLOCK BACK PASS **/
        hall(R.drawable._b317b,R.drawable._a301b,R.drawable._b301f);
        hall(R.drawable._b316b,-1,R.drawable._b317b,-1,R.drawable._b303f,-1,R.drawable._b302f);
        hall(R.drawable._b313b,-1,-1,-1,R.drawable._b316b,-1,R.drawable._b309f);
        hall(R.drawable._b312b,R.drawable._b313b,R.drawable._b313f);
        hall(R.drawable._b311b,R.drawable._b313b,R.drawable._b311f);
        hall(R.drawable._b310b,R.drawable._b311b,R.drawable._b312f);
        hall(R.drawable._b309b,-1,-1,R.drawable._b312b,R.drawable._b310b,-1,R.drawable._b314f);
        hall(R.drawable._b308b,-1,R.drawable._b309b,-1,-1,-1,R.drawable._b314f);
        hall(R.drawable._b306b,-1,-1,-1,R.drawable._b308b,-1,R.drawable._c301f);
        hall(R.drawable._b305b,-1,R.drawable._b308f,-1,R.drawable._b317b,-1,R.drawable._b303f);
        hall(R.drawable._b304b,R.drawable._b305b,R.drawable._b304f);
        hall(R.drawable._b303b,R.drawable.b303,-1,R.drawable._b304b,-1,-1,R.drawable._b305f);
        hall(R.drawable._b302b,R.drawable._b303b,R.drawable._b306f);
        hall(R.drawable._b301b,R.drawable._b302b,R.drawable._e301f);

        /** E BLOCK BACK PASS **/
        hall(R.drawable._e309b,-1,-1,R.drawable._b301b,-1,R.drawable.e302,R.drawable._e303f);
        hall(R.drawable._e308b,R.drawable._e309b,R.drawable._e304f);
        hall(R.drawable._e307b,-1,-1,R.drawable._e308b,-1,R.drawable.e303,R.drawable._e305f);
        hall(R.drawable._e306b,-1,-1,R.drawable._e307b,-1,R.drawable.e304,R.drawable._e306f);
        hall(R.drawable._e305b,R.drawable._e306b,R.drawable._e307f);
        hall(R.drawable._e304b,R.drawable._e305b,R.drawable._e308f);
        hall(R.drawable._e303b,R.drawable._e304b,R.drawable._e309f);
        hall(R.drawable._e302b,R.drawable._e303b,R.drawable._e311f);
        hall(R.drawable._e301b,R.drawable._e311f,-1,R.drawable._e303b,-1,-1,R.drawable._h301f);

        /**  H BLOCK BACK PASS **/
        hall(R.drawable._h326b,R.drawable._e301b,R.drawable._h302f);
        hall(R.drawable._h325b,-1,-1,R.drawable._h326b,-1,R.drawable._h304f,R.drawable._h330f);
        hall(R.drawable._h324b,R.drawable._h325b,R.drawable._h331f);
        hall(R.drawable._h323b,R.drawable._h324b,R.drawable._h332f);
        hall(R.drawable._h322b,R.drawable._h323b,R.drawable._h333f);
        hall(R.drawable._h321b,R.drawable._h330f,-1,-1,-1,R.drawable._h326b,R.drawable._h304f);
        hall(R.drawable._h320b,R.drawable.h302,-1,R.drawable._h321b,R.drawable.h300,-1,R.drawable._h305f);
        hall(R.drawable._h319b,-1,-1,R.drawable._h320b,-1,R.drawable.h303,R.drawable._h307f);
        hall(R.drawable._h318b,-1,R.drawable.h304,R.drawable._h319b,-1,-1,R.drawable._h308f);
        hall(R.drawable._h317b,-1,R.drawable.h306,R.drawable._h318b,-1,-1,R.drawable._h309f);
        hall(R.drawable._h316b,-1,R.drawable.h309,R.drawable._h317b,-1,-1,R.drawable._h311f);
        hall(R.drawable._h315b,-1,R.drawable._h316b,-1,R.drawable.h310,R.drawable._h313f,R.drawable._h315f);
        hall(R.drawable._h314b,R.drawable._h315b,R.drawable._h316f);
        hall(R.drawable._h313b,-1,-1,R.drawable._h314b,R.drawable.h313,-1,R.drawable._h317f);
        hall(R.drawable._h312b,-1,-1,R.drawable._h313b,R.drawable.h315,-1,R.drawable._h319f);
        hall(R.drawable._h311b,-1,R.drawable.h317,R.drawable._h312b,R.drawable.h316,R.drawable.h318,R.drawable._h322f);
        hall(R.drawable._h310b,R.drawable._h311b,R.drawable._h323f);
        hall(R.drawable._h309b,-1,-1,R.drawable._h310b,-1,R.drawable.h320,R.drawable._h324f);
        hall(R.drawable._h308b,R.drawable._h309b,R.drawable._h325f);
        hall(R.drawable._h307b,R.drawable.h322,R.drawable.h321,R.drawable._h308b,-1,-1,R.drawable._h326f);
        hall(R.drawable._h306b,-1,-1,R.drawable._h307b,R.drawable.h323,R.drawable.h324,R.drawable._h327f);
        hall(R.drawable._h305b,R.drawable._h306b,R.drawable._h328f);
        hall(R.drawable._h304b,R.drawable._h305b,R.drawable._h329f);
        hall(R.drawable._h303b,-1,R.drawable._h322b,-1,R.drawable._h304b,-1,R.drawable._h334f);
        hall(R.drawable._h302b,R.drawable._h303b,R.drawable._h335f);
        hall(R.drawable._h301b,R.drawable._h302b,R.drawable._h337f);

        /** G BLOCK BACK PASS **/
        hall(R.drawable._g309b,-1,-1,-1,R.drawable._f301f,-1,R.drawable._g302f);
        hall(R.drawable._g308b,-1,R.drawable._g309b,-1,-1,-1,R.drawable._g303f);
        hall(R.drawable._g307b,R.drawable._g308b,R.drawable._g304f);
        hall(R.drawable._g306b,-1,-1,R.drawable._g307b,-1,R.drawable.g301_1,R.drawable._g305f);
        hall(R.drawable._g305b,R.drawable._g306b,R.drawable._g306f);
        hall(R.drawable._g304b,R.drawable._g305b,R.drawable._g309f);
        hall(R.drawable._g303b,R.drawable._g304b,R.drawable._g310f);
        hall(R.drawable._g302b,-1,-1,R.drawable._g303b,-1,R.drawable.g305,R.drawable._g311f);
        hall(R.drawable._g301b,-1,-1,R.drawable._g302b,R.drawable._g313f,-1,R.drawable._g312f);

        /** F BLOCK BACK PASS **/
        hall(R.drawable._f305b,-1,-1,R.drawable._f301f,R.drawable._g301f,-1,R.drawable._f303f);
        hall(R.drawable._f304b,R.drawable._f305b,R.drawable._f304f);
        hall(R.drawable._f303b,R.drawable._f304b,R.drawable._f305f);
        hall(R.drawable._f302b,R.drawable._f303b,R.drawable._f306f);
        hall(R.drawable._f301b,-1,-1,-1,R.drawable._f302b,-1,R.drawable._f308f);

        /** C BLOCK BACK PASS **/
        hall(R.drawable._c315b,R.drawable._b306b,R.drawable._c302f);
        hall(R.drawable._c314b,-1,-1,R.drawable._c315b,-1,R.drawable.c300,R.drawable._c303f);
        hall(R.drawable._c313b,R.drawable._c314b,R.drawable._c304f);
        hall(R.drawable._c312b,-1,-1,R.drawable._c313b,-1,R.drawable.c303,R.drawable._c305f);
        hall(R.drawable._c311b,-1,-1,R.drawable._c312b,-1,R.drawable.c304,R.drawable._c306f);
        hall(R.drawable._c310b,R.drawable._c311b,R.drawable._c306f);
        hall(R.drawable._c309b,-1,R.drawable._f301b,-1,-1,-1,R.drawable._c313f);
        hall(R.drawable._c308b,-1,-1,R.drawable._c309b,-1,R.drawable.c308,R.drawable._c314f);
        hall(R.drawable._c307b,R.drawable._c308b,R.drawable._c315f);
        hall(R.drawable._c306b,R.drawable._c307b,R.drawable._c316f);
        hall(R.drawable._c305b,-1,-1,-1,R.drawable._c310b,R.drawable._c306b,R.drawable._c308f);
        hall(R.drawable._c304b,-1,-1,R.drawable._c305b,-1,R.drawable.c306_3,R.drawable._c309f);
        hall(R.drawable._c303b,R.drawable._c304b,R.drawable._c310f);
        hall(R.drawable._c302b,-1,-1,-1,R.drawable._c303b,-1,R.drawable._c311f);
        hall(R.drawable._c301b,-1,R.drawable._c302b,R.drawable.c310,-1,-1,R.drawable._c312f);

        /** D BLOCK BACK PASS **/
        hall(R.drawable._d334b,R.drawable._c301b,R.drawable._d301f);
        hall(R.drawable._d333b,R.drawable._d334b,-1,R.drawable.d300,-1,R.drawable._d313f,R.drawable._d303f);
        hall(R.drawable._d332b,R.drawable._d333b,R.drawable._d303f);
        hall(R.drawable._d331b,-1,-1,R.drawable._d304f,R.drawable._d332b,-1,R.drawable._d305f);
        hall(R.drawable._d330b,R.drawable._d331b,R.drawable._d306f);
        hall(R.drawable._d329b,R.drawable._d330b,R.drawable._d307f);
        hall(R.drawable._d328b,R.drawable._d329b,R.drawable._d308f);
        hall(R.drawable._d327b,R.drawable._d328b,R.drawable._d309f);
        hall(R.drawable._d326b,-1,-1,-1,R.drawable._d327b,-1,R.drawable._d310f);
        hall(R.drawable._d325b,-1,R.drawable._d303f,R.drawable._d334b,R.drawable.d300,R.drawable.d301,R.drawable._d314f);
        hall(R.drawable._d324b,-1,-1,R.drawable._d325b,-1,R.drawable.d303,R.drawable._d315f);
        hall(R.drawable._d323b,-1,-1,R.drawable._d324b,-1,R.drawable.d304,R.drawable._d317f);
        hall(R.drawable._d322b,R.drawable.d308,R.drawable._d318f,R.drawable._d323b,-1,-1,R.drawable._d337f);
        hall(R.drawable._d321b,R.drawable.d309_1,-1,R.drawable._d322b,-1,-1,R.drawable._d338f);
        hall(R.drawable._d320b,R.drawable.d309_2,-1,R.drawable._d321b,-1,-1,R.drawable._d340f);
        hall(R.drawable._d319b,-1,R.drawable._d326b,R.drawable._d323b,R.drawable._d336f,-1,R.drawable._d319f);
        hall(R.drawable._d318b,R.drawable._d319b,R.drawable._d320f);
        hall(R.drawable._d317b,R.drawable._d318b,R.drawable._d321f);
        hall(R.drawable._d316b,R.drawable._d317b,R.drawable._d322f);
        hall(R.drawable._d315b,R.drawable._d316b,R.drawable._d323f);
        hall(R.drawable._d314b,R.drawable._d315b,R.drawable._d323f);
        hall(R.drawable._d313b,-1,-1,R.drawable._d314b,-1,R.drawable.d319,R.drawable._d325f);
        hall(R.drawable._d312b,R.drawable._d313b,R.drawable._d327f);
        hall(R.drawable._d311b,R.drawable.d317,R.drawable.d318,-1,R.drawable._d312b,-1,R.drawable._d329f);
        hall(R.drawable._d310b,R.drawable.d316,-1,R.drawable._d311b,-1,-1,R.drawable._d330f);
        hall(R.drawable._d309b,R.drawable.d315,-1,R.drawable._d310b,-1,-1,R.drawable._d331f);
        hall(R.drawable._d308b,R.drawable.d314,-1,R.drawable._d309b,-1,-1,R.drawable._d333f);
        hall(R.drawable._d307b,R.drawable.d310,-1,R.drawable._d308b,-1,-1,R.drawable._d334f);
        hall(R.drawable._d306b,R.drawable._d307b,-1,R.drawable._d320b,-1,-1,R.drawable._d340f);
        hall(R.drawable._d305b,-1,R.drawable._d307b,R.drawable._d306b,-1,-1,R.drawable._d341f);
        hall(R.drawable._d304b,-1,R.drawable._d305b,-1,R.drawable._d341f,-1,R.drawable._d342f);
        hall(R.drawable._d303b,R.drawable._d343f,R.drawable._d343f,-1,-1,-1);
        hall(R.drawable._d302b,R.drawable._d342f,R.drawable._d306b,-1,-1,R.drawable.g310,R.drawable._j301f);
        hall(R.drawable._d301b,R.drawable._d302b,R.drawable._j301f);

        /** J BLOCK BACK PASS **/
        hall(R.drawable._j312b,-1,R.drawable._d303b,R.drawable._d301b,R.drawable._j303f,-1,R.drawable._j302f);
        hall(R.drawable._j311b,-1,R.drawable._j302f,-1,R.drawable._d301b,-1,R.drawable._j304f);
        hall(R.drawable._j310b,-1,R.drawable.j328,R.drawable._j311b,-1,R.drawable.j327,R.drawable._j305f);
        hall(R.drawable._j309b,R.drawable._j310b,R.drawable._j306f);
        hall(R.drawable._j308b,-1,-1,R.drawable._j308f,R.drawable._j309b,-1,R.drawable._j307f);
        hall(R.drawable._j307b,-1,R.drawable._j309b,R.drawable._j307f,-1,-1,R.drawable._j308f);
        hall(R.drawable._j306b,R.drawable._j307b,R.drawable._j309f);
        hall(R.drawable._j305b,-1,R.drawable._j306b,-1,-1,-1,R.drawable._j312f);
        hall(R.drawable._j304b,R.drawable._j305b,R.drawable._j313f);
        hall(R.drawable._j303b,R.drawable.j310,-1,R.drawable._j304b,-1,-1,R.drawable._j313f);
        hall(R.drawable._j302b,R.drawable._j303b,R.drawable._j314f);
        hall(R.drawable._j301b,R.drawable._j302b,R.drawable._j315f);

        /** BUTTON FIXES **/
        //hall(R.drawable._d343f,/*R.drawable._d304b*/-1,-1,-1,/*R.drawable._j312b*/-1,-1);
```

```java
        nodes[idx(R.drawable._d343f)].setOuterLeftNode(nodes[idx(R.drawable._d304b)]);
        nodes[idx(R.drawable._d343f)].setInnerRightNode(nodes[idx(R.drawable._j312b)]);
        //hall(R.drawable._d340f,-1,-1,R.drawable._d341f,-1,/*R.drawable._d307b*/-1);
        nodes[idx(R.drawable._d340f)].setOuterRightNode(nodes[idx(R.drawable._d307b)]);
        //hall(R.drawable._d339f,-1,-1,R.drawable._d340f,/*R.drawable._d307b*/-1,R.drawable.d309_2);
        nodes[idx(R.drawable._d339f)].setInnerRightNode(nodes[idx(R.drawable._d307b)]);
        //hall(R.drawable._d335f,/*R.drawable._d306b*/-1,-1,-1,-1,R.drawable._d341f);
        nodes[idx(R.drawable._d335f)].setOuterLeftNode(nodes[idx(R.drawable._d306b)]);
        //hall(R.drawable._d318f,-1,-1,R.drawable._d319f,/*R.drawable._d326b*/-1,-1);
        nodes[idx(R.drawable._d318f)].setInnerRightNode(nodes[idx(R.drawable._d326b)]);
        //hall(R.drawable._d312f,/*R.drawable._d319b*/-1,-1,-1,-1,R.drawable._d319f);
        nodes[idx(R.drawable._d312f)].setOuterLeftNode(nodes[idx(R.drawable._d319b)]);
        //hall(R.drawable._c306f,-1,-1,/*R.drawable._c306b*/-1,R.drawable._c307f,-1);
        nodes[idx(R.drawable._c306f)].setCenterNode(nodes[idx(R.drawable._c306b)]);
        //hall(R.drawable._c316f,-1,R.drawable._c306f,/*R.drawable._c310b*/-1,-1,-1);
        nodes[idx(R.drawable._c316f)].setCenterNode(nodes[idx(R.drawable._c310b)]);
        //hall(R.drawable._f301f,-1,R.drawable._f302f,-1,-1,/*R.drawable._h301b*/-1);
        nodes[idx(R.drawable._f301f)].setInnerRightNode(nodes[idx(R.drawable._h301b)]);
        //hall(R.drawable._h333f,-1,/*R.drawable._h303b*/-1,R.drawable._h334f,-1,-1);
        nodes[idx(R.drawable._h333f)].setInnerLeftNode(nodes[idx(R.drawable._h303b)]);
        //hall(R.drawable._h329f,-1,R.drawable._h334f,/*R.drawable._h303b*/-1,-1,/*R.drawable._h322b*/-1);
        nodes[idx(R.drawable._h329f)].setCenterNode(nodes[idx(R.drawable._h303b)]);
        nodes[idx(R.drawable._h329f)].setOuterRightNode(nodes[idx(R.drawable._h322b)]);

        /** TURN AROUND FIXES **/
        nodes[idx(R.drawable._a301f)].setTurnAroundNode(nodes[idx(R.drawable._a307b)]);
        nodes[idx(R.drawable._a305f)].setTurnAroundNode(nodes[idx(R.drawable._a305b)]);
        nodes[idx(R.drawable._a306f)].setTurnAroundNode(nodes[idx(R.drawable._a305b)]);

        nodes[idx(R.drawable._b307f)].setTurnAroundNode(nodes[idx(R.drawable._b302b)]);
        nodes[idx(R.drawable._b308f)].setTurnAroundNode(nodes[idx(R.drawable._b317b)]);
        nodes[idx(R.drawable._b310f)].setTurnAroundNode(nodes[idx(R.drawable._b313b)]);

        nodes[idx(R.drawable._c307f)].setTurnAroundNode(nodes[idx(R.drawable._c310b)]);

        nodes[idx(R.drawable._d302f)].setTurnAroundNode(nodes[idx(R.drawable._d334b)]);
        nodes[idx(R.drawable._d311f)].setTurnAroundNode(nodes[idx(R.drawable._d326b)]);
        nodes[idx(R.drawable._d312f)].setTurnAroundNode(nodes[idx(R.drawable._d326b)]);
        nodes[idx(R.drawable._d313f)].setTurnAroundNode(nodes[idx(R.drawable._d325b)]);
        nodes[idx(R.drawable._d316f)].setTurnAroundNode(nodes[idx(R.drawable._d324b)]);
        nodes[idx(R.drawable._d318f)].setTurnAroundNode(nodes[idx(R.drawable._d319b)]);
        nodes[idx(R.drawable._d324f)].setTurnAroundNode(nodes[idx(R.drawable._d313b)]);
        nodes[idx(R.drawable._d326f)].setTurnAroundNode(nodes[idx(R.drawable._d313b)]);
        nodes[idx(R.drawable._d328f)].setTurnAroundNode(nodes[idx(R.drawable._d312b)]);
        nodes[idx(R.drawable._d332f)].setTurnAroundNode(nodes[idx(R.drawable._d309b)]);
        nodes[idx(R.drawable._d335f)].setTurnAroundNode(nodes[idx(R.drawable._d307b)]);
        nodes[idx(R.drawable._d336f)].setTurnAroundNode(nodes[idx(R.drawable._d322b)]);
        nodes[idx(R.drawable._d339f)].setTurnAroundNode(nodes[idx(R.drawable._d306b)]);

        nodes[idx(R.drawable._e302f)].setTurnAroundNode(nodes[idx(R.drawable._e309b)]);
        nodes[idx(R.drawable._e310f)].setTurnAroundNode(nodes[idx(R.drawable._e303b)]);

        nodes[idx(R.drawable._f302f)].setTurnAroundNode(nodes[idx(R.drawable._f305b)]);
        nodes[idx(R.drawable._f307f)].setTurnAroundNode(nodes[idx(R.drawable._f302b)]);

        nodes[idx(R.drawable._g307f)].setTurnAroundNode(nodes[idx(R.drawable._g305b)]);
        nodes[idx(R.drawable._g308f)].setTurnAroundNode(nodes[idx(R.drawable._g304b)]);
        nodes[idx(R.drawable._g313f)].setTurnAroundNode(nodes[idx(R.drawable._g301b)]);

        nodes[idx(R.drawable._h303f)].setTurnAroundNode(nodes[idx(R.drawable._h326b)]);
        nodes[idx(R.drawable._h306f)].setTurnAroundNode(nodes[idx(R.drawable._h320b)]);
        nodes[idx(R.drawable._h310f)].setTurnAroundNode(nodes[idx(R.drawable._h317b)]);
        nodes[idx(R.drawable._h312f)].setTurnAroundNode(nodes[idx(R.drawable._h316b)]);
        nodes[idx(R.drawable._h318f)].setTurnAroundNode(nodes[idx(R.drawable._h313b)]);
        nodes[idx(R.drawable._h320f)].setTurnAroundNode(nodes[idx(R.drawable._h311b)]);
        nodes[idx(R.drawable._h321f)].setTurnAroundNode(nodes[idx(R.drawable._h311b)]);
        nodes[idx(R.drawable._h336f)].setTurnAroundNode(nodes[idx(R.drawable._h302b)]);
```

```
      nodes[idx(R.drawable._h338f)].setTurnAroundNode(nodes[idx(R.drawable._h301b)]);

      nodes[idx(R.drawable._j303f)].setTurnAroundNode(nodes[idx(R.drawable._j311b)]);
      nodes[idx(R.drawable._j310f)].setTurnAroundNode(nodes[idx(R.drawable._j306b)]);
      nodes[idx(R.drawable._j311f)].setTurnAroundNode(nodes[idx(R.drawable._j305b)]);
    }
  }

```

```java
/**
 * TourHall.java
 * Brock Butler
 * A hallway node in the tour. Hallways have multiple branching points,
 * and can turn around. Each hallway knows about all of it's branching points.
 * Created by Taras Mychaskiw 2013-02-20
 * Copyright (c) 2013 Sea Addicts. All rights reserved.
 */
package edu.seaaddicts.brockbutler.tour;

import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Toast;
import edu.seaaddicts.brockbutler.R;

public class TourHall extends TourNode {
  protected TourNode[] nodes; //each of the nodes this node branches off to

  /**
   * Forward pass constructor. Defines each button, and inits the turn around
   * location to null.
   * @param img - the image resource value
   * @param ll - outer left node
   * @param ul - upper left node
   * @param c - center node
   * @param ur - upper right node
   * @param lr - outer right node
   */
  public TourHall(int img, TourNode ll, TourNode ul, TourNode c, TourNode ur, TourNode lr){
    image = img;
    nodes = new TourNode[5];
    nodes[0] = ll;
    nodes[1] = ul;
    nodes[2] = c;
    nodes[3] = ur;
    nodes[4] = lr;
    turnAroundNode = null;
    title = makeTitle(img);
  }

  /**
   * Second pass constructor. Defines each button, and also defines the turn
   * around node. Links `turnAroundNode` back to this node via turning around.
   * @param ta - turn around node
   */
  public TourHall(int img, TourNode ll, TourNode ul, TourNode c, TourNode ur, TourNode lr, TourNode ta){
    this(img,ll,ul,c,ur,lr);
    turnAroundNode = ta;
    turnAroundNode.setTurnAroundNode(this);
  }

  public void setOuterLeftNode(TourNode node){  nodes[0] = node; }
  public void setInnerLeftNode(TourNode node){  nodes[1] = node; }
  public void setCenterNode(TourNode node){     nodes[2] = node; }
  public void setInnerRightNode(TourNode node){ nodes[3] = node; }
  public void setOuterRightNode(TourNode node){ nodes[4] = node; }

  /**
   * @param img - the resource value of the image of this node
   * @return - the value for `title`
   */
  private String makeTitle(int img){
    switch (img){
    case R.drawable.b303: return("B303");
    case R.drawable.c306_3: return("C306");
    case R.drawable.g301_1: return("G301");
    case R.drawable.h304: return("H304");
    case R.drawable.h306: return("H306");
    case R.drawable.h306a6: case R.drawable.h306a9: return("H306A");
    case R.drawable.h309: return("H309");
    case R.drawable.h309a6: case R.drawable.h309a9: return("H309A");
    case R.drawable.h318: return("H318");
    case R.drawable._a301f: case R.drawable._a301b:
      return("A Block");
    case R.drawable._b301f: case R.drawable._b301b: case R.drawable._b306b:
      return("B Block");
    case R.drawable._c301f: case R.drawable._c301b: case R.drawable._c313f:
      return("C Block");
    case R.drawable._d301f: case R.drawable._d301b: case R.drawable._d304b:
      return("D Block");
    case R.drawable._e301f: case R.drawable._e301b:
      return("E Block");
    case R.drawable._f301b: case R.drawable._f302f:
      return("F Block");
    case R.drawable._g301f:
      return("G Block");
    case R.drawable._h301f: case R.drawable._h301b:
      return("H Block");
    case R.drawable._j301f: case R.drawable._j312b:
      return("J Block");
    }
    return(null);
  }

  /**
   * Changes the image displayed on the screen and redefines where the buttons lead us to.
   * Also pushes this node onto the TourInfo's history.
   */
  @Override
  public void paint(final TourInfo info){
    info.rl.setBackgroundResource(image);
    info.current = this;
    if (title != null)
      Toast.makeText(info.context,title,Toast.LENGTH_SHORT).show();
    for (int i = 0; i < nodes.length; i++){
      final int idx = i;
      info.buttons[idx].setOnClickListener(new OnClickListener(){
        public void onClick(View v){
          if (nodes[idx] != null){
            info.history.push(TourHall.this);
            nodes[idx].paint(info);
          }
        }
      });
      int resID = (nodes[idx] == null)? R.drawable.qb_empty : info.arrows[idx];
      info.buttons[idx].setImageResource(resID);
    }
  }
}
```

```java
/**
 * TourInfo.java
 * Brock Butler
 * Wrapper class for all passing required information around throughout the tour.
 * Created by Taras Mychaskiw 2013-02-20
 * Copyright (c) 2013 Sea Addicts. All rights reserved.
 */
package edu.seaaddicts.brockbutler.tour;

import java.util.Stack;

import android.content.Context;
import android.widget.ImageButton;
import android.widget.RelativeLayout;
import edu.seaaddicts.brockbutler.R;

public class TourInfo {
  public RelativeLayout rl;
  public ImageButton[] buttons;
  public int[] arrows;
  public Context context;
  public TourNode current;
  public Stack<TourNode> history;

  /**
   * @param r - the layout of which the background is changed from node to node
   * @param b - array of ImageButtons where one would tap to change nodes
   * @param c - getApplicationContext(), for toasts
   */
  public TourInfo(RelativeLayout r, ImageButton[] b, Context c){
    rl = r;
    buttons = b;
    context = c;
    current = null;
    history = new Stack<TourNode>();
    arrows = new int[5];
    arrows[0] = R.drawable.qb_outerleft;
    arrows[1] = R.drawable.qb_innerleft;
    arrows[2] = R.drawable.qb_center;
    arrows[3] = R.drawable.qb_innerright;
    arrows[4] = R.drawable.qb_outerright;
  }
}

```

```java
/**
 * TourNode.java
 * Brock Butler
 * Simple abstract wrapper for TourRoom and TourHall.
 * Created by Taras Mychaskiw 2013-02-20
 * Copyright (c) 2013 Sea Addicts. All rights reserved.
 */
package edu.seaaddicts.brockbutler.tour;

public abstract class TourNode {
  protected int image;    //resource for the image on this node
  protected TourNode turnAroundNode;
  protected String title;   //text for Toast to display, if any

  public boolean canTurnAround(){ return(turnAroundNode != null); }
  public TourNode getTurnAroundNode(){ return(turnAroundNode); }

  public void setTurnAroundNode(TourNode node){ turnAroundNode = node; }

  public void setOuterLeftNode(TourNode node){}
  public void setInnerLeftNode(TourNode node){}
  public void setCenterNode(TourNode node){}
  public void setInnerRightNode(TourNode node){}
  public void setOuterRightNode(TourNode node){}

  /**
   * Changes the background image and redefines what the buttons do,
   * ie where the buttons will take you.
   * @param info - current tour state
   */
  public abstract void paint(TourInfo info);
}

```

```java
/**
 * TourRoom.java
 * Brock Butler
 * A room node in the tour. Rooms are simple, they don't go anywhere.
 * Created by Taras Mychaskiw 2013-02-20
 * Copyright (c) 2013 Sea Addicts. All rights reserved.
 */
package edu.seaaddicts.brockbutler.tour;

import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Toast;
import edu.seaaddicts.brockbutler.R;

public class TourRoom extends TourNode {

  /**
   * Initializes this room, rooms only need the image resource value
   * @param img - the resource value of the image of this room
   */
  public TourRoom(int img, String s){
    image = img;
    title = s;
    turnAroundNode = null;
  }

  @Override
  public void paint(TourInfo info) {
    info.rl.setBackgroundResource(image);
    info.current = this;
    if (title != null)  //should never be null, but just in case
      Toast.makeText(info.context,title,Toast.LENGTH_SHORT).show();
    for (int i = 0; i < info.buttons.length; i++){
      info.buttons[i].setOnClickListener(new OnClickListener(){
        public void onClick(View v){}
      });
      info.buttons[i].setImageResource(R.drawable.qb_empty);
    }
  }
}

```

```
<menu xmlns:android="http://schemas.android.com/apk/res/android" >

    <item
        android:id="@+id/menu_go_back"
        android:onClick="goBack"
        android:title="@string/menu_go_back"
        android:titleCondensed="@string/menu_go_back">
    </item>
    <item
        android:id="@+id/menu_turn_around"
        android:onClick="turnAround"
        android:title="@string/menu_turn_around"
        android:titleCondensed="@string/menu_turn_around">
    </item>
    <item
        android:id="@+id/menu_teleport"
        android:title="@string/menu_teleport"
        android:titleCondensed="@string/menu_teleport">
        <menu>
            <item
                android:id="@+id/teleport_a_block"
                android:onClick="teleport"
                android:title="@string/a_block"
                android:titleCondensed="@string/a_block"/>
            <item
                android:id="@+id/teleport_b_block"
                android:onClick="teleport"
                android:title="@string/b_block"
                android:titleCondensed="@string/b_block"/>
            <item
                android:id="@+id/teleport_c_block"
                android:onClick="teleport"
                android:title="@string/c_block"
                android:titleCondensed="@string/c_block"/>
            <item
                android:id="@+id/teleport_d_block"
                android:onClick="teleport"
                android:title="@string/d_block"
                android:titleCondensed="@string/d_block"/>
            <item
                android:id="@+id/teleport_e_block"
                android:onClick="teleport"
                android:title="@string/e_block"
                android:titleCondensed="@string/e_block"/>
            <item
                android:id="@+id/teleport_f_block"
                android:onClick="teleport"
                android:title="@string/f_block"
                android:titleCondensed="@string/f_block"/>
            <item
                android:id="@+id/teleport_g_block"
                android:onClick="teleport"
                android:title="@string/g_block"
                android:titleCondensed="@string/g_block"/>
            <item
                android:id="@+id/teleport_h_block"
                android:onClick="teleport"
                android:title="@string/h_block"
                android:titleCondensed="@string/h_block"/>
            <item
                android:id="@+id/teleport_j_block"
                android:onClick="teleport"
                android:title="@string/j_block"
                android:titleCondensed="@string/j_block"/>
        </menu>
    </item>
    <item
        android:id="@+id/menu_end_tour"
        android:onClick="endTour"
        android:title="@string/menu_end_tour"
        android:titleCondensed="@string/menu_end_tour">
    </item>

</menu>
```

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/screen"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    tools:context=".TourHandler" >

    <ImageButton
        android:id="@+id/outerleft"
        android:contentDescription="@string/go_left"
        android:background="@android:color/transparent"
        android:src="@drawable/qb_outerleft"
        android:padding="5dp"
        android:scaleType="fitEnd"
        android:layout_width="75dp"
        android:layout_height="400dp"
        android:layout_alignParentLeft="true"
        android:layout_marginLeft="10dp"
        android:layout_centerVertical="true" />

    <ImageButton
        android:id="@+id/innerleft"
        android:contentDescription="@string/go_left"
        android:background="@android:color/transparent"
        android:src="@drawable/qb_innerleft"
        android:padding="5dp"
        android:scaleType="fitEnd"
        android:layout_width="75dp"
        android:layout_height="400dp"
        android:layout_marginLeft="20dp"
        android:layout_toRightOf="@+id/outerleft"
        android:layout_centerVertical="true" />

    <ImageButton
        android:id="@+id/center"
        android:contentDescription="@string/go_forward"
        android:background="@android:color/transparent"
        android:src="@drawable/qb_center"
        android:padding="5dp"
        android:scaleType="fitEnd"
        android:layout_width="75dp"
        android:layout_height="400dp"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true" />

    <ImageButton
        android:id="@+id/innerright"
        android:contentDescription="@string/go_right"
        android:background="@android:color/transparent"
        android:src="@drawable/qb_innerright"
        android:padding="5dp"
        android:scaleType="fitEnd"
        android:layout_width="75dp"
        android:layout_height="400dp"
        android:layout_marginRight="20dp"
        android:layout_toLeftOf="@+id/outerright"
        android:layout_centerVertical="true" />

    <ImageButton
        android:id="@+id/outerright"
        android:contentDescription="@string/go_right"
        android:background="@android:color/transparent"
        android:src="@drawable/qb_outerright"
        android:padding="5dp"
        android:scaleType="fitEnd"
        android:layout_width="75dp"
        android:layout_height="400dp"
        android:layout_alignParentRight="true"
        android:layout_marginRight="10dp"
        android:layout_centerVertical="true" />
```

```

</RelativeLayout>

```