**Project 5 – Graphics Programming**

**(20 points)**

**Name:** Ozaner Hansha

**Due Date: Friday, December 4, 2015**

**Description:**

We have been learning about the ACM Graphics library and Event-Driven Programming lately. In this project, you will get to try out several ideas. Be sure to include Javadoc and write clean, commented, nicely formatted code.

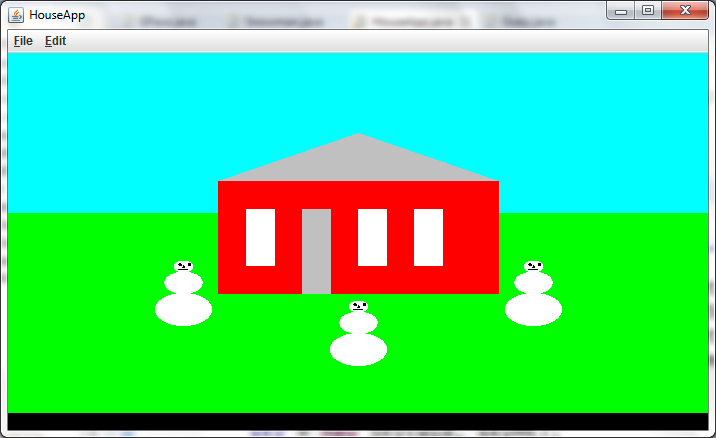
Carry out the following steps:

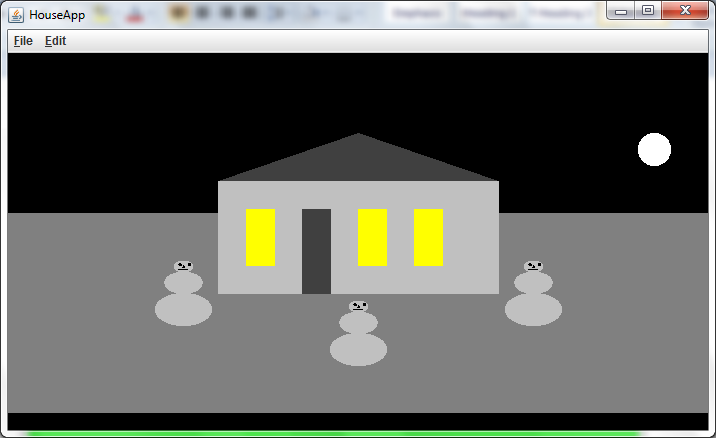
1. Create a House class, which is a subclass of GCompound. Your house must have at least three windows, a door and a roof. The house color must be something other than white, black or gray. The roof and door are light gray. The windows should be white. Beyond that, elaborate as you want.
2. Create a Snowman class, which is a subclass of GCompound. Your snowman must have at least three body parts, eyes, a nose and a mouth. The snowman should be white. Beyond that it can be as elaborate as you want. You can define additional classes for the face, etc. if you want, but it isn’t required.
3. Create a Sky class, which is a subclass of GCompound. The sky is blue.
4. Create a Grass class, which is a subclass of GCompound. The grass is green.
5. Create a Street class, which is a subclass of GCompound. The street is black.
6. Create a HouseApp class, which is a subclass of GraphicsProgram.
7. In the init method of HouseApp, create a scene and place a house and three snowmen in the grassy yard (which is green). The street is in front of the house and is black. The grass-sky horizon is behind (not over) the house. Beyond that, elaborate as you want. (Perhaps a driveway?)
8. Add a mouseClicked event that turns day into night and vice-versa. At night,
   1. the sky turns black
   2. the moon appears in the sky
   3. the windows turn yellow
   4. the roof and door turn dark gray
   5. the house color turns a darker shade or light gray
   6. the snowmen turn light gray
   7. the grass turns a darker shade or gray

To complete the project, submit this Word document as an attachment to a message to the class Hand-in folder with:

* your name filled out above
* the source listings for HouseApp, House, Snowman, Sky, Grass and Street (each starting on a new page)
* a screen shot of the day scene, a screen shot of the night scene

Sample Scenes





Here is a template for the HouseApp to get you started:

**import** acm.program.\*;

**import** java.awt.event.\*;

@SuppressWarnings("serial")

**public** **class** HouseApp **extends** GraphicsProgram {

/\*\* Set the program dimensions. \*/

**public** **static** **final** **int** *APPLICATION\_WIDTH* = 700;

**public** **static** **final** **int** *APPLICATION\_HEIGHT* = 400;

**boolean** day = **true**;

Street street;

House house;

Grass grass;

Sky sky;

Snowman snowman1, snowman2, snowman3;

// this allows the program to run as an application

**public** **static** **void** main(String[] args) {

**new** HouseApp().start(args);

}

**public** **void** init() {

setSize(*APPLICATION\_WIDTH*, *APPLICATION\_HEIGHT*);

// make objects and add them to the canvas

addMouseListeners();

}

**public** **void** mouseClicked(MouseEvent e) {

// change day to night by invoking a method for each object to change its appearance

// ...

// house.setLight(...);

// ...

}

}

Here is a template for the house. Feel free to change it as needed.

**import** java.awt.Color;

**import** acm.graphics.\*;

@SuppressWarnings("serial")

**public** **class** House **extends** GCompound {

// define constants as you need them

// scale your components as percentages of your house width and height

// declare your component objects (roof, door, windows, etc.)

**public** House(**double** wid, **double** ht) {

// make a roof, door, windows, etc.

// ...

// add them to the GCompound

}

**public** **void** setLight(**boolean** day) {

// change the properties of the component objects based on the time of day

}

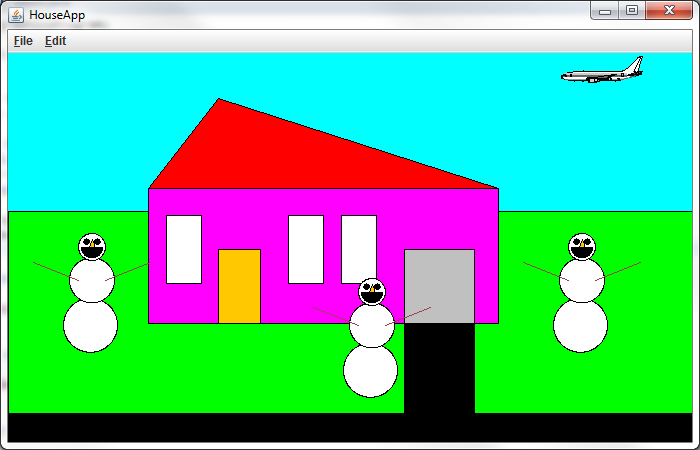
}

**Extensions**

For those who get done early, try this.

1. Fly a plane across the sky in the region of sky above the housetop in the day time. After it leaves the screen, send it across the sky again. It would work best if you can find an image with a transparent background.
2. Send Santa and his sleigh across the sky at night.

***Screenshots:***





***HouseApp Class:***

**package** unit4.graphicsProgram;

**import** java.awt.Dimension;

**import** java.awt.event.MouseEvent;

**import** acm.graphics.GCompound;

**import** acm.program.GraphicsProgram;

/\*\*

\* This is the Main Class of the Graphics Program Project.<br>

\* This program creates a scene that changes when it is clicked.<br><br>

\*

\* AP Computer Science - Pd. 7<br>

\* December 3rd, 2015<br>

\* Dr. Jones<br>

\* **@author** Ozaner Hansha

\*/

@SuppressWarnings("serial")

**public** **class** HouseApp **extends** GraphicsProgram

{

/\*\*

\* The width and height of the application.

\*/

**public** **static** **final** Dimension ***APP\_SIZE*** = **new** Dimension(700,560);

/\*\*

\* Whether or not the program is displaying daytime. If false then it is nighttime.

\*/

**public** **boolean** isDay = **true**;

/\*\*

\* An array of all objects in the scene.

\*/

**public** Object[] objects =

{

**new** Sky(),**new** Grass(),**new** Street()

};

/\*\*

\* Places all objects in the Graphics Scene.

\* **@param** x - Width of the program window.

\* **@param** y - Height of the program window.

\*/

**public** HouseApp(**int** x, **int** y)

{

setSize(x,y);

**for**(Object g: objects)

{

add((GCompound)g);

}

}

/\* Adds Mouse Listener

\* @see acm.program.GraphicsProgram#init()

\*/

@Override

**public** **void** init()

{

addMouseListeners();

}

/\* Sets size of window.

\* @see acm.program.GraphicsProgram#run()

\*/

@Override

**public** **void** run()

{

setSize((**int**)***APP\_SIZE***.getWidth(),(**int**)***APP\_SIZE***.getHeight());

}

/\* Switches light status of all objects.

\* @see acm.program.Program#mouseClicked(java.awt.event.MouseEvent)

\*/

@Override

**public** **void** mouseClicked(MouseEvent e)

{

isDay = !isDay;

**for**(Object g: objects)

{

((Nighttimeable)g).setLight(isDay);

}

}

/\*\*

\* The main method of the {@link HouseApp} program.<br>

\* Creates a new Graphics Scene.

\* **@param** args - Not expecting any command line input.

\*/

**public** **static** **void** main(String[] args)

{

HouseApp app = **new** HouseApp((**int**)***APP\_SIZE***.getWidth(),(**int**)***APP\_SIZE***.getHeight());

app.start();

}

}

***Nighttimeable Interface:***

**package** unit4.graphicsProgram;

/\*\*

\* This interface allows all objects to share common functionality,

\* in regards to their day/night cycle.

\* **@author** Ozaner Hansha

\*/

**public** **interface** Nighttimeable

{

/\*\*

\* Calls the {@link #daytime()} if day is true and {@link #nighttime()} if not.

\* **@param** day - Whether or not is is day.

\*/

**public** **default** **void** setLight(**boolean** day)

{

**if**(day)

daytime();

**else**

nighttime();

}

/\*\*

\* Sets the object to its daytime status.

\*/

**public** **abstract** **void** nighttime();

/\*\*

\* Sets the object to its nighttime status.

\*/

**public** **abstract** **void** daytime();

}

***House Class:***

**package** unit4.graphicsProgram;

**import** java.awt.Color;

**import** acm.graphics.GCompound;

**import** acm.graphics.GPolygon;

**import** acm.graphics.GRect;

/\*\*

\* This class creates house objects based on {@link GCompound}.

\* **@author** Ozaner Hansha

\*/

@SuppressWarnings("serial")

**public** **class** House **extends** GCompound **implements** Nighttimeable

{

**private** GPolygon roof;

**private** GRect body, door, garage, driveway;

**private** GRect[] windows = **new** GRect[3];

/\*\*

\* Creates a house with a red roof, magenta body, white windows, gray garage, and black driveway.

\* **@param** x - The x coordinate of the top left of the house's base.

\* **@param** y - The y coordinate of the top left of the house's base.

\*/

**public** House(**double** x, **double** y)

{

//Roof

roof = **new** GPolygon(x,y);

roof.addVertex(0, 0);

roof.addEdge(HouseApp.***APP\_SIZE***.getWidth()\*.1, -.2\*HouseApp.***APP\_SIZE***.getHeight());

roof.addVertex(HouseApp.***APP\_SIZE***.getWidth()\*.5, 0);

roof.setFilled(**true**);

add(roof);

//Body

body = **new** GRect(x,y,HouseApp.***APP\_SIZE***.getWidth()\*.5,HouseApp.***APP\_SIZE***.getHeight()\*.3);

body.setFilled(**true**);

add(body);

//Door

door = **new** GRect(body.getWidth()\*.12,body.getHeight()\*.55);

door.setFilled(**true**);

add(door,body.getX()+body.getWidth()\*.2,body.getY()+body.getHeight()-door.getHeight());

//Windows

**double** windowY = body.getY()+body.getHeight()\*.2;

**double** windowWidth = body.getWidth()\*.1, windowHeight = body.getHeight()\*.5;

windows[0] = **new** GRect(body.getX()+body.getWidth()\*.05,windowY,windowWidth,windowHeight);

windows[1] = **new** GRect(body.getX()+body.getWidth()\*.4,windowY,windowWidth,windowHeight);

windows[2] = **new** GRect(body.getX()+body.getWidth()\*.55,windowY,windowWidth,windowHeight);

**for**(GRect g: windows)

{

g.setFilled(**true**);

add(g);

}

//Garage

garage = **new** GRect(body.getWidth()\*.2,body.getHeight()\*.55);

garage.setFilled(**true**);

add(garage,body.getX()+body.getWidth()\*.73,body.getY()+body.getHeight()-garage.getHeight());

//Driveway

driveway = **new** GRect(body.getWidth()\*.2,body.getHeight()\*.7);

driveway.setFilled(**true**);

add(driveway,garage.getX(),garage.getY()+garage.getHeight());

daytime(); //init Compound

}

/\* Sets the roof, door, and garage dark gray. Sets the body light gray.

\* Sets all the windows yellow.

\* @see unit4.graphicsProgram.Nighttimeable#nighttime()

\*/

@Override

**public** **void** nighttime()

{

roof.setFillColor(Color.***DARK\_GRAY***);

body.setFillColor(Color.***LIGHT\_GRAY***);

door.setFillColor(Color.***DARK\_GRAY***);

garage.setFillColor(Color.***DARK\_GRAY***);

**for**(GRect g: windows)

{

g.setFillColor(Color.***YELLOW***);

}

}

/\* Sets the roof red, the body magenta, the door orange,

\* the garage light gray, and all the windows white.

\* @see unit4.graphicsProgram.Nighttimeable#daytime()

\*/

@Override

**public** **void** daytime()

{

roof.setFillColor(Color.***RED***);

body.setFillColor(Color.***MAGENTA***);

door.setFillColor(Color.***ORANGE***);

garage.setFillColor(Color.***LIGHT\_GRAY***);

**for**(GRect g: windows)

{

g.setFillColor(Color.***WHITE***);

}

}

}

***Grass Class:***

**package** unit4.graphicsProgram;

**import** java.awt.Color;

**import** java.util.Timer;

**import** java.util.TimerTask;

**import** acm.graphics.GCompound;

**import** acm.graphics.GImage;

**import** acm.graphics.GRect;

/\*\*

\* This class creates the grass and the objects placed on it.

\* It is based on {@link GCompound}.

\* **@author** Ozaner Hansha

\*/

@SuppressWarnings("serial")

**public** **class** Grass **extends** GCompound **implements** Nighttimeable

{

**private** GRect grass;

**private** Snowman[] snowmen = **new** Snowman[3];

**private** House house;

**private** GImage planeTakeOff;

/\*\*

\* Creates a green rectangle, a {@link House}, 3 {@link Snowman}, and a gif of a plane taking off.

\*/

**public** Grass()

{

//Grass Background

grass = **new** GRect(0,HouseApp.***APP\_SIZE***.getHeight()\*.35,

HouseApp.***APP\_SIZE***.getWidth(),HouseApp.***APP\_SIZE***.getHeight()\*.5);

grass.setFilled(**true**);

add(grass);

//House

house = **new** House(HouseApp.***APP\_SIZE***.getWidth()\*.2,HouseApp.***APP\_SIZE***.getHeight()\*.3);

add(house);

//Plane take off

planeTakeOff = **new** GImage("planeTakeOff.gif");

add(planeTakeOff,HouseApp.***APP\_SIZE***.getWidth()-planeTakeOff.getWidth(),0);

//3 Snowmen

snowmen[0] = **new** Snowman(HouseApp.***APP\_SIZE***.getWidth()\*.1,HouseApp.***APP\_SIZE***.getHeight()\*.4); //Left Snowman

snowmen[1] = **new** Snowman(HouseApp.***APP\_SIZE***.getWidth()\*.5,HouseApp.***APP\_SIZE***.getHeight()\*.5); //Middle Snowman

snowmen[2] = **new** Snowman(HouseApp.***APP\_SIZE***.getWidth()\*.8,HouseApp.***APP\_SIZE***.getHeight()\*.4); //Right Snowman

**for**(Snowman s: snowmen)

add(s);

planeTakeOffAnimation(); //Starts plane take off animation

daytime(); //init compound

}

/\*\*

\* Sets the {@link planeTakeOff} object invisible after 2.5 seconds.

\*/

**public** **void** planeTakeOffAnimation()

{

**new** Timer().schedule(**new** TimerTask()

{

@Override

**public** **void** run()

{

planeTakeOff.setVisible(**false**);

}

}, 2500);

}

/\* Sets the grass gray, and the house and snowman to nighttime.

\* @see unit4.graphicsProgram.Nighttimeable#nighttime()

\*/

@Override

**public** **void** nighttime()

{

grass.setFillColor(Color.***GRAY***);

house.nighttime();

**for**(Snowman s: snowmen)

s.nighttime();

}

/\* Sets the grass green, and the house and snowman to daytime.

\* @see unit4.graphicsProgram.Nighttimeable#daytime()

\*/

@Override

**public** **void** daytime()

{

grass.setFillColor(Color.***GREEN***);

house.daytime();

**for**(Snowman s: snowmen)

s.daytime();

}

}

***Sky Class:***

**package** unit4.graphicsProgram;

**import** java.awt.Color;

**import** acm.graphics.GCompound;

**import** acm.graphics.GRect;

/\*\*

\* This class creates the sky and the objects placed in it.

\* It is based on {@link GCompound}.

\* **@author** Ozaner Hansha

\*/

@SuppressWarnings("serial")

**public** **class** Sky **extends** GCompound **implements** Nighttimeable

{

**private** GRect sky;

**private** FlyingObject flyingObj;

**private** Moon moon;

/\*\*

\* Creates the a blue rectangle, a {@link Moon}, and a {@link FlyingObject}.

\*/

**public** Sky()

{

//Sky

sky = **new** GRect(0,0,HouseApp.***APP\_SIZE***.getWidth(),HouseApp.***APP\_SIZE***.getHeight()\*.5);

sky.setFilled(**true**);

add(sky);

//Moon

moon = **new** Moon(HouseApp.***APP\_SIZE***.getWidth()\*.8,HouseApp.***APP\_SIZE***.getHeight()\*.1);

add(moon);

//Airplane-Santa

flyingObj = **new** FlyingObject();

add(flyingObj);

daytime(); //init compound

}

/\* Sets the sky black, and the flying object and moon to nighttime.

\* @see unit4.graphicsProgram.Nighttimeable#nighttime()

\*/

@Override

**public** **void** nighttime()

{

sky.setColor(Color.***BLACK***);

flyingObj.setLight(**false**);

moon.setLight(**false**);

}

/\* Sets the sky cyan, and the flying object and moon to daytime.

\* @see unit4.graphicsProgram.Nighttimeable#daytime()

\*/

@Override

**public** **void** daytime()

{

sky.setColor(Color.***CYAN***);

flyingObj.setLight(**true**);

moon.setLight(**true**);

}

}

***Street Class:***

**package** unit4.graphicsProgram;

**import** java.awt.Color;

**import** acm.graphics.GCompound;

**import** acm.graphics.GRect;

/\*\*

\* This class creates the street and is based on {@link GCompound}.

\* **@author** Ozaner Hansha

\*/

@SuppressWarnings("serial")

**public** **class** Street **extends** GCompound **implements** Nighttimeable

{

**private** GRect street;

/\*\*

\* Creates a black rectangle at the bottom of the screen.

\*/

**public** Street()

{

street = **new** GRect(0,HouseApp.***APP\_SIZE***.getHeight()\*.8,

HouseApp.***APP\_SIZE***.getWidth(),HouseApp.***APP\_SIZE***.getHeight()\*.1);

street.setFilled(**true**);

street.setColor(Color.***BLACK***);

add(street);

}

/\* Doesn't do anything.

\* @see unit4.graphicsProgram.Nighttimeable#nighttime()

\*/

@Override

**public** **void** nighttime(){}

/\* Doesn't do anything.

\* @see unit4.graphicsProgram.Nighttimeable#daytime()

\*/

@Override

**public** **void** daytime(){}

}

***Moon Class:***

**package** unit4.graphicsProgram;

**import** java.awt.Color;

**import** acm.graphics.GCompound;

**import** acm.graphics.GOval;

/\*\*

\* This class creates a moon that has multiple phases that change each night.

\* **@author** Ozaner Hansha

\*/

@SuppressWarnings("serial")

**public** **class** Moon **extends** GCompound **implements** Nighttimeable

{

**private** GOval moon, illumination;

**private** **int** phase = -10;

/\*\*

\* Creates a white circle and a black one on top of it.

\* **@param** x - The x coordinate of the top left corner of the moon.

\* **@param** y - The x coordinate of the top left corner of the moon.

\*/

**public** Moon(**double** x, **double** y)

{

moon = **new** GOval(HouseApp.***APP\_SIZE***.getWidth()\*.1,HouseApp.***APP\_SIZE***.getWidth()\*.1);

moon.setFilled(**true**);

moon.setFillColor(Color.***WHITE***);

moon.setVisible(**false**);

add(moon,x,y);

illumination = **new** GOval(moon.getWidth(),moon.getHeight());

illumination.setFilled(**true**);

illumination.setFillColor(Color.***BLACK***);

illumination.setVisible(**false**);

add(illumination,moon.getX()+moon.getWidth(),y);

}

/\* Sets the moon and illumination visible and increases the phase of the moon.

\* @see unit4.graphicsProgram.Nighttimeable#nighttime()

\*/

@Override

**public** **void** nighttime()

{

illumination.setLocation(moon.getX()+illumination.getWidth()\*.1\*phase, illumination.getY());

**if**(phase > -10)

phase--;

**else**

phase = 10;

moon.setVisible(**true**);

illumination.setVisible(**true**);

}

/\* Sets the moon and illumination invisible.

\* @see unit4.graphicsProgram.Nighttimeable#daytime()

\*/

@Override

**public** **void** daytime()

{

moon.setVisible(**false**);

illumination.setVisible(**false**);

}

}

***Snowman Class:***

**package** unit4.graphicsProgram;

**import** java.awt.Color;

**import** acm.graphics.GArc;

**import** acm.graphics.GCompound;

**import** acm.graphics.GLine;

**import** acm.graphics.GOval;

/\*\*

\* This class creates snowman objects based on {@link GCompound}.

\* **@author** Ozaner Hansha

\*/

@SuppressWarnings("serial")

**public** **class** Snowman **extends** GCompound **implements** Nighttimeable

{

**private** GOval[] body = **new** GOval[3];

**private** GOval eyeLeft, eyeRight;

**private** GArc mouth, nose;

**private** GLine armLeft = **new** GLine(0,0,0,0), armRight = **new** GLine(0,0,0,0);

/\*\*

\* Constructs a snowman with 3 white ovals, an orange nose, a black mouth, and 2 brown arms.

\*/

**public** Snowman(**double** x, **double** y)

{

//Body

body[2] = **new** GOval(x,y,HouseApp.***APP\_SIZE***.getHeight()\*.06,HouseApp.***APP\_SIZE***.getHeight()\*.06); //Head

body[1] = **new** GOval(x-body[2].getWidth()/3,body[2].getY()+body[2].getHeight()\*.9,

HouseApp.***APP\_SIZE***.getHeight()\*.1,HouseApp.***APP\_SIZE***.getHeight()\*.1); //Middle

body[0] = **new** GOval(x-body[1].getWidth()/3,body[1].getY()+body[1].getHeight()\*.9,

HouseApp.***APP\_SIZE***.getHeight()\*.12,HouseApp.***APP\_SIZE***.getHeight()\*.12); //End

**for**(GOval g: body)

{

g.setFilled(**true**);

g.setFillColor(Color.***WHITE***);

add(g);

}

//Eyes

**double** eyeY = body[2].getY()+body[2].getHeight()\*.2;

**double** eyeWidth = body[2].getWidth()\*.2, eyeHeight = body[2].getHeight()\*.2;

eyeLeft = **new** GOval(body[2].getX()+body[2].getWidth()\*.2,eyeY,eyeWidth,eyeHeight);

eyeLeft.setFilled(**true**);

add(eyeLeft);

eyeRight = **new** GOval(body[2].getX()+body[2].getWidth()\*.6,eyeY,eyeWidth,eyeHeight);

eyeRight.setFilled(**true**);

add(eyeRight);

//Mouth

mouth = **new** GArc(body[2].getWidth()\*.8,body[2].getHeight()\*.8,0,0);

mouth.setFilled(**true**);

add(mouth);

//Nose

nose = **new** GArc(body[2].getWidth()\*.4,body[2].getHeight()\*.6,0,60);

nose.setFilled(**true**);

add(nose, body[2].getX()+body[2].getWidth()\*.2,body[2].getY()+body[2].getHeight()\*.2);

//Arms

add(armLeft);

add(armRight);

daytime(); //init compound

}

/\* Turns eyes red, makes mouth frown,

\* @see unit4.graphicsProgram.Nighttimeable#nighttime()

\*/

@Override

**public** **void** nighttime()

{

**for**(GOval g: body)

{

g.setFillColor(Color.***LIGHT\_GRAY***);

}

eyeLeft.setFillColor(Color.***RED***);

eyeRight.setFillColor(Color.***RED***);

mouth.setLocation(body[2].getX()+body[2].getWidth()\*.1,body[2].getY()+body[2].getHeight()\*.43);

mouth.setSweepAngle(180);

nose.setFillColor(Color.***DARK\_GRAY***);

//Arms

**double** armStartY = body[1].getY()+body[1].getHeight()\*.5;

**double** armEndY = body[1].getY()+body[1].getHeight();

armLeft.setColor(Color.***BLACK***);

armLeft.setStartPoint(body[1].getX()+body[1].getWidth()\*.8,armStartY);

armLeft.setEndPoint(body[1].getX()+body[1].getWidth()\*1.8,armEndY);

armRight.setColor(Color.***BLACK***);

armRight.setStartPoint(body[1].getX()+body[1].getWidth()\*.2,armStartY);

armRight.setEndPoint(body[1].getX()+body[1].getWidth()\*-.8,armEndY);

}

/\* Turns eyes black, makes mouth smile,

\* @see unit4.graphicsProgram.Nighttimeable#nighttime()

\*/

@Override

**public** **void** daytime()

{

**for**(GOval g: body)

{

g.setFillColor(Color.***WHITE***);

}

eyeLeft.setFillColor(Color.***BLACK***);

eyeRight.setFillColor(Color.***BLACK***);

mouth.setLocation(body[2].getX()+body[2].getWidth()\*.1,body[2].getY()+body[2].getHeight()\*.1);

mouth.setSweepAngle(-180);

nose.setFillColor(Color.***ORANGE***);

Color brown = **new** Color(156, 65, 82);

**double** armStartY = body[1].getY()+body[1].getHeight()\*.5;

**double** armEndY = body[1].getY()+body[1].getHeight()\*.1;

armLeft.setColor(brown);

armLeft.setStartPoint(body[1].getX()+body[1].getWidth()\*.8,armStartY);

armLeft.setEndPoint(body[1].getX()+body[1].getWidth()\*1.8,armEndY);

armRight.setColor(brown);

armRight.setStartPoint(body[1].getX()+body[1].getWidth()\*.2,armStartY);

armRight.setEndPoint(body[1].getX()+body[1].getWidth()\*-.8,armEndY);

}

}

***FlyingObject Class (Airplane/Santa):***

**package** unit4.graphicsProgram;

**import** java.util.Timer;

**import** java.util.TimerTask;

**import** acm.graphics.GCompound;

**import** acm.graphics.GImage;

@SuppressWarnings("serial")

**public** **class** FlyingObject **extends** GCompound **implements** Nighttimeable

{

**private** GImage airplane = **new** GImage("airplaneFlying.png"), santa = **new** GImage("santaFlying.gif");

/\*\*

\* Creates 2 images, and sets them to move across the screen.

\*/

**public** FlyingObject()

{

//Construct GCompound

add(airplane);

add(santa);

createAnimationThread();

daytime(); //init compound

}

/\*\*

\* Creates a new {@link Timer} with a {@link TimerTask} that plays the object's animation.

\*/

**private** **void** createAnimationThread()

{

GCompound temp = **this**;

**new** Timer().scheduleAtFixedRate(**new** TimerTask()

{

@Override

**public** **void** run()

{

**while**(temp.getX() + temp.getWidth() > 0)

{

temp.move(-10, 0);

pause(75);

}

setLocation(HouseApp.***APP\_SIZE***.getWidth(),0);

}

}, 3000, 2000);

setLocation(HouseApp.***APP\_SIZE***.getWidth(),0);

}

/\* Sets the airplane invisible and santa visible.

\* @see unit4.graphicsProgram.Nighttimeable#nighttime()

\*/

@Override

**public** **void** nighttime()

{

airplane.setVisible(**false**);

santa.setVisible(**true**);

}

/\* Sets the airplane visible and santa invisible.

\* @see unit4.graphicsProgram.Nighttimeable#daytime()

\*/

@Override

**public** **void** daytime()

{

airplane.setVisible(**true**);

santa.setVisible(**false**);

}

}