This exam will cover material from Chapter 6, 7, and 8

Here are some general topics for the exam from those chapters:
- **method overloading**
- **calling a superclass constructor from subclass**
- **static and final**
- **java.util.List objects**
- **for-each loop**
- **intersecting objects (using getOneIntersectingObject method)**
- **using GreenfootImages and painting our own images**
- **for-loops**
- **animation (explosions and smoke examples from class)**
- **using the mouse to capture user interactions**

(1) What is overloading (specifically method overloading)? When is it used?
Method overloading is having two methods with the same name that differ in number and/or type of parameters.

(2) When are static and final used in a variable declaration?
Static and final appear when we want to make the variable a constant (the value of the variable will never change.)

(3) Write the code to see if an actor is colliding with another (an Obstacle). If there is a collision, the image of the Obstacle should change to "nothreat.png".
Actor obstacle = getOneIntersectingObject(Obstacle.class);
if(obstacle != null) {
   obstacle.setImage("nothreat.png");
}

//You don't need to create a variable to hold the intersected obstacles, but then the code needs to look like this:
if(getOneIntersectingObject(Obstacle.class) != null) {
   getOneIntersectingObject(Obstacle.class).setImage("nothreat.png");
}

(4) Write the code to create a GreenfootImage with a blue background and five white squares at random locations. The squares should be sized 20x20.
GreenfootImage image = new GreenfootImage(500,500);
image.setColor(java.awt.Color.BLUE);
image.fill();
image.setColor(java.awt.Color.WHITE);
int x;
int y;
for(int count = 0; count < 5; count++) {
    x = Greenfoot.getRandomNumber(500);
    y = Greenfoot.getRandomNumber(500);
    image.fillRect(x, y, 20, 20);
}

(5) Use the following for-loop definition to answer parts a – d.
```
for(int count = 1; count < 9; count++) {
    addObject(new Question(), 34, 34);
}
```

   a) What is the initial value of this loop's counter variable?
      1

   b) What is the value of this loop's counter variable when the loop is done executing?
      9

   c) Circle the part of the code above that is considered the loop body.
      Highlighted in yellow.

   d) How many times would this loop execute?
      8

(6) Which of the following would be the correct choice to fill in the blank in the code to make this loop execute 5 times?
```
for (int count = 1; _____; count++) {
    //some code for loop
}
```
   a) count < 5
   b) count <= 5
   c) count < 6
   d) count <= 6

(7) Write the code that gets all the Flowers from the scenario and then moves each flower 5 pixels to its right.
```
java.util.List<Flower> flowers = getWorld().getObjects(Flower.class);

for(Flower f: flowers) {
    f.setLocation(f.getX() +5, f.getY());
}
```

(8) Write the code so that an actor will follow the movements of the mouse on the world.
```
MouseInfo mouse = Greenfoot.getMouseInfo();

if(mouse != null) {
    setLocation(mouse.getX(), mouse.getY());
}
```

(9) Reproduce the code in the grow() or shrink() methods that we discussed in class that goes through an array of pre-defined images and sets the image of the actor to the next image of the series.
```
if(_imageCount < 30) {
    setImage(_images[_imageCount]);
    _imageCount++;
}
else {
    getWorld().removeObject(this);
}
```