CSE 113 B

October 4 – 8, 2010

Announcements

- Lab 5 posted
- Exam 2 – Friday 10/15; Review sheet available Friday 10/8

3

Piano Example

- Make a piano that the user can play using the keyboard (Chapter 5 of text).
- First, we made the keys react when the user clicked a key on the keyboard.

4

Abstraction and the Key Class

- We added parameters to the constructor of the Key class and an instance variable so we didn't need 12 different Key classes to make the keys on our piano.
- We simply created 12 key objects and input different parameters as we created them.

5

Positioning Keys

- Next, we needed to make 12 keys in the world.
- It would be nice to have the work done by the computer instead of us having to write the code to insert 12 keys by copying and pasting.

6

Loops

- Repetition in programs allows us to repeat something over and over.
- We achieve repetition through loops.
- We will look at a while loop to help us repeat.

7

WHILE-LOOP

- This will keep looping until the condition indicated on the loop is false.

```
while (booleanExpression)
{
    //code that should be repeated
}
```

8

WHILE-LOOPS

```
while (true)
{
    //code that should be repeated
}
```

- Remember that we said that this loop will continue forever because true is always true.

9

While-Loop

- In order to help us keep track of how many times we are looping, we need to create a variable to store a count.
- Inside the loop, we also must remember to increment the count so that the loop executes the correct number of times.

10

While-Loops

```
int count = 0;
while (count < 10)
{
        //code that should be repeated
        count = count + 1;
}
```

- The code in this loop will execute 10 times
- Remember that there were a number of places that starting at 0 for counting helped us in this example.

11

Piano Keys

- Placing them at random locations doesn't seem right.
- So, we placed them all at the same y-coordinate and used the loop counter to help us position each key's x-value so that they lined up all in a row.

12

Piano Key reactions

- Still all react to the same key on the keyboard, so we need a way to list out all of the keys that we want the user's to press on the keyboard and then we can assign the keyboard keys to the piano keys as we create the piano keys in our program.

13

Arrays

- A type of collection (way to keep track of a group of objects).
- Arrays are fixed size.
- To declare a variable that holds an array:

  TypeOfThingInArray[] name;
- To put things into the array:

  name = {thing1, thing2, thing3… thingn};
  - Where thingx are the actual values stored in the array.

14

Arrays

- To create a new, empty array and assign it to the variable:

  name = new TypeOfThingInArray[NUMBER];
  - Where number is the number of elements you can store in the array.
- Note: We didn't do the above in class, but this is still a viable way to create an array.

15

Arrays

- You can access elements in a array by using their index.
- Indices for an array are from 0 to size -1. So, if there are 20 elements in an array, valid indices are 0-19.

```
nameOfArray[index]
```

- Would allow you to access the element at that index

```
nameOfArray[index] = blah;
```

- Would assign blah to that index.

16

Additional Boolean Operations

- Can help us create more complex boolean expressions for inside () for if-statements or loops.
- And (&&)
  - Conjunction – true only when both conjuncts are true.
- Or (||)
  - Disjunction – false only when both disjuncts are false.
- Not (!)
  - Negation – changes the truth value between false and true.

17

Movement and Animation

- Actually controlling the actor on our own as opposed to using move(), turn(), and others from a superclass.

18

Move & Rotate

- We moved the actor by finding its current position and changing it just slightly.
- We can change the rotation of the image of the actor by calling setRotation.
- But, we noticed that simply changing the rotation of the image does not make the movement correct.

19

Movement

- In move method (from Vehicle): I do not expect you to be able to derive formulas to create movement taking into account rotation.
- We are going to discuss movement using a Vector next week – you will be responsible for using that in movement.

20

Check Edges

- In checkEdges method: We needed to get the width of the world, so we first needed to get the world and then needed to ask the world for its width.

getWorld().getWidth()

a world object

ask the world object for its width