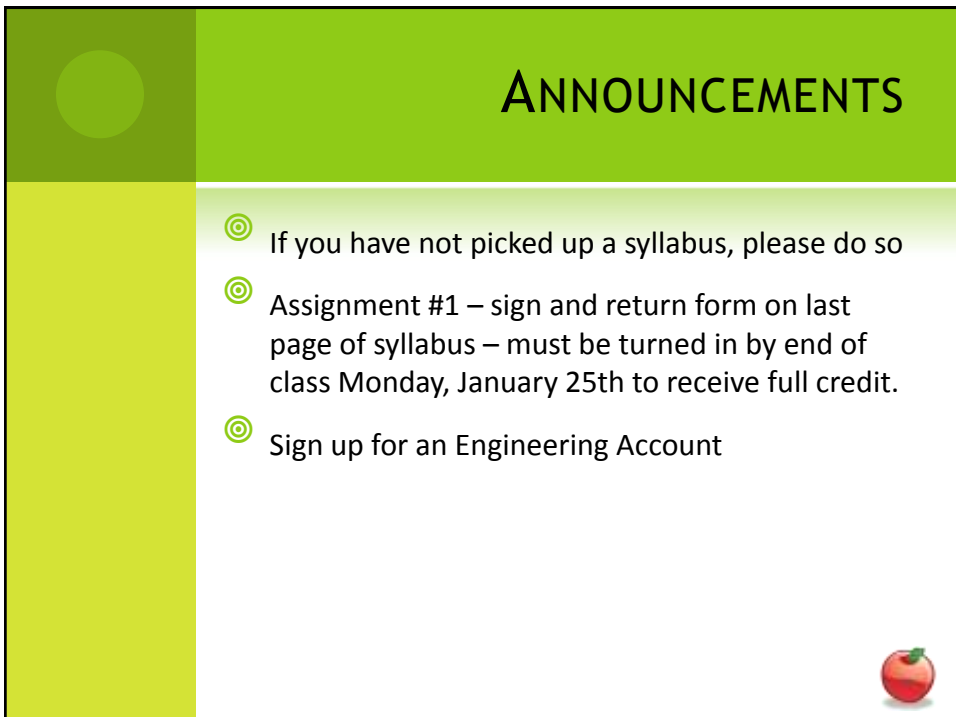



CSE 113 B

January 18 – 22, 2010



ANNOUNCEMENTS

- ⊙ If you have not picked up a syllabus, please do so
- ⊙ Assignment #1 – sign and return form on last page of syllabus – must be turned in by end of class Monday, January 25th to receive full credit.
- ⊙ Sign up for an Engineering Account



3

Class [Programming Term]

-definition of some thing
-description

Objects

↳ "thing"

The actual thing that does something in the program.



The screenshot shows the Greenfoot IDE interface. The main workspace is a grid labeled 'wombatWorld'. On the right side, there is a 'Scenario Information' panel showing a class hierarchy: 'World' (parent of 'WombatWorld'), 'Actor' (parent of 'Wombat' and 'Leaf'), and 'Wombat' (parent of 'Leaf'). Handwritten annotations include:

- 'World' with an arrow pointing to the 'World' class in the hierarchy.
- 'Class diagram panel' with an arrow pointing to the entire hierarchy.
- 'class box' with an arrow pointing to the 'Leaf' class box.
- 'Execution Controls' with an arrow pointing to the 'Go', 'Run', and 'Reset' buttons at the bottom.

5

```


inherited from Object ▶
inherited from Actor ▶
void act()
boolean canMove()
void eatLeaf()
boolean foundLeaf()
int getLeavesEaten()
void move()
void setDirection(int direction)
void turnLeft()
Inspect
Remove

```

List of methods

Formal specification of an action [PW]

When we select, we are "calling" or "invoking" a method



6

Method:


return type method name (parameter list)

Return type: type of information returned after a method is executed

void - means nothing is returned

boolean - true or false is returned

int - integer (whole) number



7

Method name

- selected by programmer

Rules:

- first character must be a letter
- method names can only contain letters, digits & underscores

a_b



8

Parameter List

- Enclosed by ()
- Can be empty
- If it's not empty, it contains parameters

↳ additional information needed to perform the method/action.



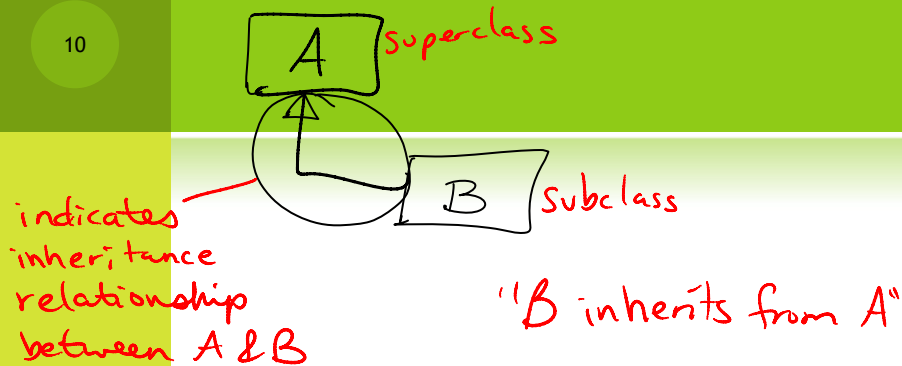
9

METHODS RECAP

- ⊙ Three parts:
 - ⊙ Return type – type of information that is returned from a method call; void if nothing returned
 - ⊙ Name – the name given to the method by the programmer
 - ⊙ Parameters – information that is needed to perform the actions of the method; empty () if none needed
- ⊙ Methods can have a return type, but no parameters; can have parameters, but no return type; can have no return type and no parameters; can have a return type and parameters



10



B inherits all of A's methods, but B will typically also define its own unique methods

