

# CSE 142 Final Cheat Sheet

```

for (initialization; test; update) {
    statement(s);
    ...
}

if (test) {
    statement(s);
} else if (test) {
    statement(s);
} else {
    statement(s);
}

type name = value; // variable declaration and initialization

Type objectName = new Type(parameters); // object construction
    
```

Math Method	Description
Math.abs( <i>value</i> )	absolute value
Math.min( <i>v1</i> , <i>v2</i> )	smaller of two values
Math.max( <i>v1</i> , <i>v2</i> )	larger of two values
Math.round( <i>value</i> )	nearest whole number
Math.sqrt( <i>value</i> )	square root
Math.pow( <i>b</i> , <i>e</i> )	base to the exponent power

Random Method	Description
nextInt( <i>max</i> )	random integer from 0 to <i>max</i> -1

String Method	Description
contains( <i>str</i> )	true if this string contains the other's characters inside it
endsWith( <i>str</i> ), startsWith( <i>str</i> )	true if this string starts/ends with the other's characters
equals( <i>str</i> )	true if this string is the same as <i>str</i>
equalsIgnoreCase( <i>str</i> )	true if this string is the same as <i>str</i> , ignoring capitalization
indexOf( <i>str</i> )	index in this string where given string begins (-1 if not found)
length()	number of characters in this string
replace( <i>str1</i> , <i>str2</i> )	replace all occurrences in this string of <i>str1</i> with <i>str2</i>
substring( <i>i</i> , <i>j</i> )	characters in this string from index <i>i</i> (inclusive) to <i>j</i> (exclusive)
toLowerCase(), toUpperCase()	a new string with all lowercase or uppercase letters
charAt( <i>i</i> )	returns char at index <i>i</i>

Scanner Method	Description
nextInt()	reads/returns input token as <code>int</code>
next()	reads/returns input token as <code>String</code>
nextDouble()	reads/returns input token as <code>double</code>
nextLine()	reads/returns line as <code>String</code>
hasNextInt()	returns <code>true</code> if there is a next token and it can be read as an <code>int</code>
hasNext()	returns <code>true</code> if there is a next token to read
hasNextDouble()	returns <code>true</code> if there is a next token and it can be read as a <code>double</code>
hasNextLine()	returns <code>true</code> if there is a next line to read

## Declaring and using Arrays

```
type[] name = new type[length];  
type[] name = {VAL1, VAL2, VAL3, ...};  
name[index] = value;
```

## Critter classes

```
public class name extends Critter {  
    fields  
  
    constructor  
  
    public Color getColor() {  
        statement(s) that return a Color;  
    }  
  
    public Action getMove() {  
        statement(s) that return an Action  
    }  
  
    public String toString() {  
        statement(s) that return a String;  
    }  
}
```

## Critter constants

Actions	Directions	Neighbors
Action.HOP	Direction.NORTH	Neighbor.SAME
Action.INFECT	Direction.SOUTH	Neighbor.OTHER
Action.RIGHT	Direction.EAST	Neighbor.WALL
Action.LEFT	Direction.WEST	Neighbor.EMPTY

CritterInfo Method	Description
getFront()	returns neighbor in front of you
getBack()	returns neighbor in back of you
getLeft()	returns neighbor to your left
getRight()	returns neighbor to your right
getDirection()	returns direction you are facing

## Declaring and using ArrayLists

```
ArrayList<Type> name = new ArrayList<Type>();
```

ArrayList Method	Description
add( <b>value</b> )	appends value at end of list
add( <b>index, value</b> )	inserts value at index, shifting subsequent values right
clear()	removes all elements of the list
contains( <b>value</b> )	whether the list contains the value
get( <b>index</b> )	returns the value at given index
isEmpty()	whether list is empty
remove( <b>index</b> )	removes value at index, shifting subsequent values left
set( <b>index, value</b> )	replaces value at given index with given value
size()	returns the size of the list