You have already seen nested loops - for example with a command tree like:

```
seq
  x=1;
while
  x<5
  seq
    y=2;
    while
      y<=4
      print(x,y,x*y);
      y=y+1;
    seq
      x=x+1;
```

The Java code is on the next page.
```java
{x
    x=1; // Init outer loop var
    while(x<5) {
        y=2; // Init inner loop var
        while(y<=4) {
            out.print(""+x+','+y+','+(x*y)); // Inner Loop Body
            y=y+1; // Inner advance
        }
        x=x+1; // Outer advance
    }
}
```
```java
{ 
    x=1;           #1
    while(x<5) {   #2
        y=2;       #3
        while(y<=4) {  #4
            out.print(""+x+','+y+','+(x*y));  #5
            y=y+1;      #6
        }
        x=x+1;      #7
    }
}
```
Like in While Loop Histories

Homework:
Note state vector is the two loop variables (x,y)
```java
{ 
    x=1;          #1
    while (x<5) { #2
        y=2;     #3
        while (y<=4) { #4
            out.print(""+x+',','+y+',','+(x*y));  #5
            y=y+1;        #6
        }
        x=x+1;      #7
    }
}
```

Note Loop produces EVERY combination of
Values of
    x in 1,2,3,4
With
    y in 2,3,4

It is kind of like a "multiplication"

<table>
<thead>
<tr>
<th>(x,y)</th>
<th>y=2</th>
<th>y=3</th>
<th>y=4</th>
</tr>
</thead>
<tbody>
<tr>
<td>x=1</td>
<td>(1,2)</td>
<td>(1,3)</td>
<td>(1,4)</td>
</tr>
<tr>
<td>x=2</td>
<td>(2,2)</td>
<td>(2,3)</td>
<td>(2,4)</td>
</tr>
<tr>
<td>x=3</td>
<td>(3,2)</td>
<td>(3,3)</td>
<td>(3,4)</td>
</tr>
<tr>
<td>x=4</td>
<td>(4,2)</td>
<td>(4,3)</td>
<td>(4,4)</td>
</tr>
</tbody>
</table>

In fact, it is called the Cartesian product by mathematicians
The previous loop is called "Free Running" because every given value of x is combined with every given value of y.

Below is a loop that is NOT free running - it is called "interlinked"
This loop is expressed using the arithmetic for instead of the while loop - although there is not real difference between the two

```java
for(x=1;x<=3;x=x+1) {
    for(y=2*x;y<=3*x;y=y+1) {
        out.format("pair(%d,%d)\n",x,y);
    }
}
```

However, below is exactly the same thing written using while loops and a println instead of a format.

```java
x=1; while(x<=3) {
    y=2*x; while(y<=3*x) {
        out.println("pair("+x+","+y+")");
        y = y+1;
    }
    y = x+1;
}
```
Below is a table showing the paired values of $x$ and $y$ actually used for output.

<table>
<thead>
<tr>
<th>$x$</th>
<th>$y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
</tr>
</tbody>
</table>

This is strange because the range of values for the inner loop variable $y$ depends on the value of the outer loop variable $x$.

When $x$ is 1, the range for $y$ is $2x$ through $3x$, namely 2 through 3.

But when $x$ is 2, the range for $y$ is $2x$ through $3x$ which is now 4 through 6.

Finally when $x$ is 3, the range for $y$ is $2x$ through $3x$ which this time is 6 through 9.
The kinds of loops used in nesting are completely up to your choice - while loop, arithmetic for loop, for-in loop, etc.

Here is an interlinked example with an outer loop ranging over an int[] and the inner loop ranging over an arithmetic series:

```java
int[] data = { 3,5,2,6 }; for(int x : data) {
    for(y=0; y<x ; y=y+2 ) {
        out.format("pair(%d,%d)\n",x,y);
    }
} x y
3 0
3 2
5 0
5 2
5 4
2 0
6 0
6 2
6 4
```

**Watch Out**

NOTE: x and y have no special meaning - you could used any TWO loop variables.
As yet another example:

```java
int[][] data = { {3,5} , {2} , {9,1,4,5} };
for( int[] x : data ) {
    for(int y : x ) {
        out.format("pair(%d,%d)\n",x,y);
    }
}
```

Note type of outer loop variable `x` is NOT `int` but rather `int[]` - namely array of integers.

Note "trick" - value of outer loop variable `x` becomes the range for the inner loop variable `y`.

Values of `(x,y)` paired by this loop:

<table>
<thead>
<tr>
<th><code>x</code></th>
<th><code>y</code></th>
</tr>
</thead>
<tbody>
<tr>
<td>[3,5]</td>
<td>3</td>
</tr>
<tr>
<td>[3,5]</td>
<td>5</td>
</tr>
<tr>
<td>[2]</td>
<td>2</td>
</tr>
<tr>
<td>[9,1,4,5]</td>
<td>9</td>
</tr>
<tr>
<td>[9,1,4,5]</td>
<td>1</td>
</tr>
<tr>
<td>[9,1,4,5]</td>
<td>4</td>
</tr>
<tr>
<td>[9,1,4,5]</td>
<td>5</td>
</tr>
</tbody>
</table>