

Table 1

Pointers	
recursion code	
<pre>public static IntList incrList(IntList L, int x) {     if (L == null) {         return L;     }     IntList incr = new IntList(L.first + x, incrList(L.rest, x));     return incr; }</pre>	
CS61B lists1/exercises/ExtraIntListPractice.java - iterative method	
	<p><b>new</b> creates a box with 64bits- address in it, which points to the object with default value; When we <b>instantiate</b> an Object (e.g. Dog, Walrus, Planet): Java first allocates a box of bits for each instance variable of the class and fills them with a default value (e.g. 0, null). parameter passing: copy the bits into the new scope</p>
<pre>public static void main(String[] args) {     IntList L = new IntList(15, null);     L = new IntList(10, L);     L = new IntList(5, L); }</pre>	<div>FramesObjects</div> <div><div><div>&lt;init&gt;:5</div><div>this</div><div>f15</div><div>rnull</div></div><div><div>IntList instance</div><div><div>first0</div><div>restnull</div></div></div></div>
constructor	after going through the constructor: The constructor then usually fills every such box with some other value.
<pre>public IntList(int f, IntList r) {     first = f;     rest = r; }</pre>	<div>FramesObjects</div> <div><div><div>&lt;init&gt;:8</div><div>this</div><div>f15</div><div>rnull</div><div>Return valuevoid</div></div><div><div>IntList instance</div><div><div>first15</div><div>restnull</div></div></div></div>
<pre>public static void main(String[] args) {     IntList L = new IntList(15, null);     L = new IntList(10, L);     L = new IntList(5, L);      IntList N = IntList.incrList2(L, 1);     System.out.println(L.get_Recur(0)); }</pre>	<div>FramesObjects</div> <div><div><div>incrList2:52</div><div>M</div><div>x1</div></div><div><div>main:80</div><div>L</div></div><div><div>IntList instance</div><div><div>first5</div><div>rest</div></div><div><div>IntList instance</div><div><div>first10</div><div>rest</div></div><div><div>IntList instance</div><div><div>first15</div><div>restnull</div></div></div></div></div></div>
	pointer temp
<pre>public static IntList incrList2(IntList M, int x) {     if (M == null) {         return M;     }     IntList temp = M;     IntList incr = new IntList(temp.first + x, null);     IntList temp2 = incr;     while (temp.rest != null) {         temp = temp.rest;         temp2.rest = new IntList(temp.first + x, temp2.rest);     }     return incr; }</pre>	<div>FramesObjects</div> <div><div><div>incrList2:56</div><div>M</div><div>x1</div><div>temp</div></div><div><div>main:80</div><div>L</div></div><div><div>IntList instance</div><div><div>first5</div><div>rest</div></div><div><div>IntList instance</div><div><div>first10</div><div>rest</div></div><div><div>IntList instance</div><div><div>first15</div><div>restnull</div></div></div></div></div></div>
创建new IntList object	

<pre>public static IntList incrList2(IntList M, int x) {     if (M == null) {         return M;     }     IntList temp = M;     IntList incr = new IntList(temp.first + x, null);     IntList temp2 = incr;     while (temp.rest != null) {         temp = temp.rest;         temp2.rest = new IntList(temp.first + x, null);         temp2 = temp2.rest;     }     return incr; }</pre>	<div>Frames<div>incrList2:57<div>M<div>•</div></div><div>x<div>1</div></div><div>temp<div>•</div></div><div>incr<div>•</div></div></div><div>main:80<div>L<div>•</div></div></div></div> <div>Objects<div>IntList instance<div>first<div>5</div></div><div>rest<div>•</div></div></div><div>IntList instance<div>first<div>10</div></div><div>rest<div>•</div></div></div><div>IntList instance<div>first<div>15</div></div><div>rest<div>null</div></div></div><div>IntList instance<div>first<div>6</div></div><div>rest<div>null</div></div></div></div>