CS 125 - Lecture 32

Objectives: Recursion; activation diagrams;

Up next: Quiz this week; MP6 out - due ??;

3. **Find a word in the list; return link:**

```
public Link find( String searchFor) {




}
```

5a. **Create an activation diagram for f1(200):**

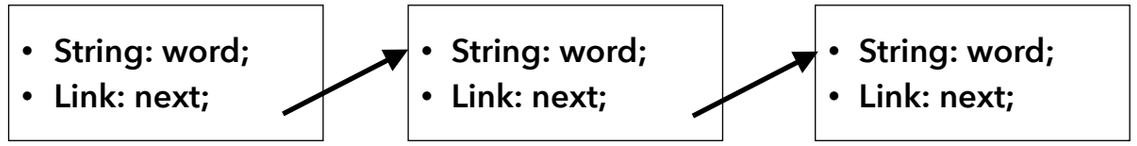
```
static int f1(int c) {
   if (c <= 25) return 7;
   int x = c / 2;
   return f1(x) + f1(x);
}
```

5b. **Will f2(200) return the same result as f1(200)?**

Create an activation diagram for f2(200):

```
static int f2(int c) {
   if (c <= 25) return 7;
   int x = c / 2;
   return 2 * f2(x);
}
```

Linked List



1. **Write a recursive instance method to return a string with all the words concatenated together:**

2. **Write a recursive instance method that returns a reference to the last link:**

```
public Link getLastLink() {
   if(next == null) // BASE CASE



   else



}
```

```
    public void addToEnd(Link newLink) {









     }
```

6. **If each link has a larger value than the previous, will the following** `getMax()` **create a tree or chain of activations?**

```
class LinkedList{
   int value;
   LinkedList next;
}
public int getMax() {
   if (next == null)
      return value;  // BASE CASE

   int result = next.getMax();
   if (result < value) return value;
   else return next.getMax();
}
```

7a. **Create an activation diagram on the right for** `prc(3, "*", true)`:

```
public static void prc(int c, String s, boolean newline) {
    if (newline && c==0) {
       System.out.println();
       return;
    }
    System.out.print(s);
    prc(c-1, s, newline);
}
```

7b. **How many stars are printed for:** `prc(3, "*", false)`?

8. **Create an activation diagram for** `f3(31373)`:

```
public static int f3(int x) {
   if (x == 3) return 1;
   if (x < 10) return 0;

   return f3(x/10) + f3(x%10);
}
```

9. **Discuss with a neighbor your favorite desktop application's graphical user interface. Why do you like it? How does it help facilitate using the application?**

10. **Desktop user interfaces in Java:**

- Composed of graphical elements or components
- Elements are all objects: properties, behavior, inheritance

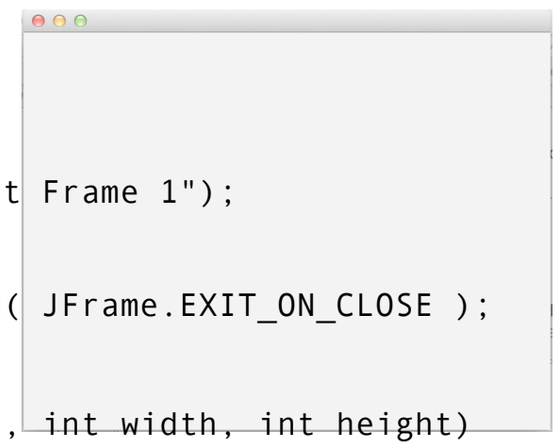
11. **JFrame**

- Subclass of Container;
- Defines a rectangular area on screen to hold components (graphical objects like buttons, sliders, text labels, etc.)
- To use, import graphics packages:

```
import java.awt.*
import javax.swing.*
```

- Usage:

```
JFrame frame = new JFrame("Test Frame 1");
frame.setSize(200,100);
frame.setVisible( true );
frame.setDefaultCloseOperation( JFrame.EXIT_ON_CLOSE );
```



docs (hover):

```
public void setBounds(int x, int y, int width, int height)
```

3. **What is a graphical user interface?**

4. **Discuss with a neighbor your favorite desktop application's graphical user interface. Why do you like it? How does it help facilitate using the application?**

5. **Desktop user interfaces in Java:**

- Composed of graphical elements or components
- Elements are all objects: properties, behavior, inheritance

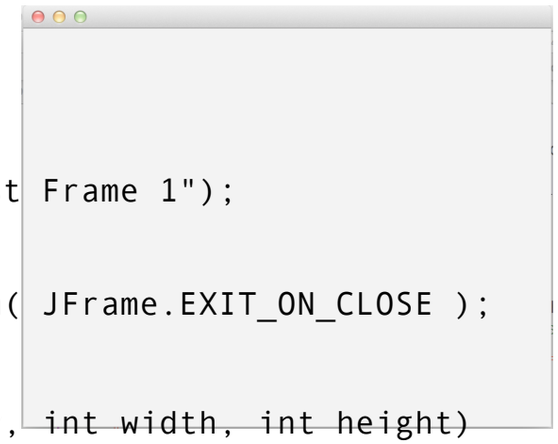
6. **JFrame**

- Subclass of Container;
- Defines a rectangular area on screen to hold components (graphical objects like buttons, sliders, text labels, etc.)
- To use, import graphics packages:

```
import java.awt.*
import javax.swing.*
```

- Usage:

```
JFrame frame = new JFrame("Test Frame 1");
frame.setSize(200,100);
frame.setVisible( true );
frame.setDefaultCloseOperation( JFrame.EXIT_ON_CLOSE );
```



docs (hover):

```
public void setBounds(int x, int y, int width, int height)
```

7a. **Create an activation diagram for f1(200):**

```
static int f1(int c) {
   if (c <= 25) return 7;
   int x = c / 2;
   return f1(x) + f1(x);
}
```

7b. **Will f2(200) return the same result as f1(200)?**

Create an activation diagram for f2(200):

```
static int f2(int c) {
   if (c <= 25) return 7;
   int x = c / 2;
   return 2 * f2(x);
}
```

8a. **Create an activation diagram on the right for** `prc(3, "*", true)`**:**

```
public static void prc(int c, String s, boolean newline)
{
    if (newline && c==0) {
       System.out.println();
       return;
    }
    System.out.print(s);
    prc(c-1, s, newline);
}
```

8b. **How many stars are printed for:** `prc(3, "*", false)`**?**