Recursion

Michael Wagner

Recursive Case

```
public class Recursion
{
    public static void printAll(List<String> stuff, int subscript)
    {
        if (subscript < stuff.size())
        {
            System.out.println(stuff.get(subscript));
            printAll(stuff, ++subscript);
        }
    }

    public static void main(String[] args)
    {
        List<String> top_sites = new ArrayList<String>();

        top_sites.add("google.com");
        top_sites.add("youtube.com");
        top_sites.add("facebook.com");
        top_sites.add("msn.com");

        printAll(top_sites, 0);
    }
}
```

Example

Using recursion, print a list in reverse. Step through the algorithm using the debugger.

Base Case and Recursive Case

```
public class Recursion
{
    public static int sum(List<Integer> stuff, int subscript)
    {
        if (subscript < stuff.size())
        {
            return stuff.get(subscript) + sum(stuff, ++subscript);
        }
        else
        {
            return 0;
        }
    }

    public static void main(String[] args)
    {
        List<Integer> number_list = new ArrayList<Integer>();

        number_list.add(50);
        number_list.add(60);
        number_list.add(40);
        number_list.add(900);

        int result = sum(number_list, 0);

        System.out.println(result);
    }
}
```

Example

Using recursion, calculate the factorial of a number

Classes and Recursion

```
public class Website
{
    private String[] accounts = {"elon@tesla.com",
                                 "mark@facebook.com",
                                 "sergei@google.com"};

    public boolean contains(String needle, int subscript)
    {
        if (subscript == needle.length())
        {
            return false;
        }
        else if (accounts[subscript].equals(needle))
        {
            return true;
        }
        else
        {
            return contains(needle, ++subscript);
        }
    }
}
```

Convenience

```
public class Website
{
    private String[] accounts = {"elon@tesla.com",
                                 "mark@facebook.com",
                                 "sergei@google.com"};

    public boolean contains(String needle, int subscript)
    {
        if (subscript == needle.length())
        {
            return false;
        }
        else if (accounts[subscript].equals(needle))
        {
            return true;
        }
        else
        {
            return contains(needle, ++subscript);
        }
    }

    public boolean contains(String needle)
    {
        return contains(needle, 0);
    }
}
```

Example

Make a class called **BruteForce**. Make a constructor that accepts an integer **max**. Create a recursive method that prints all strings of length **max** that contains only A-B-C.