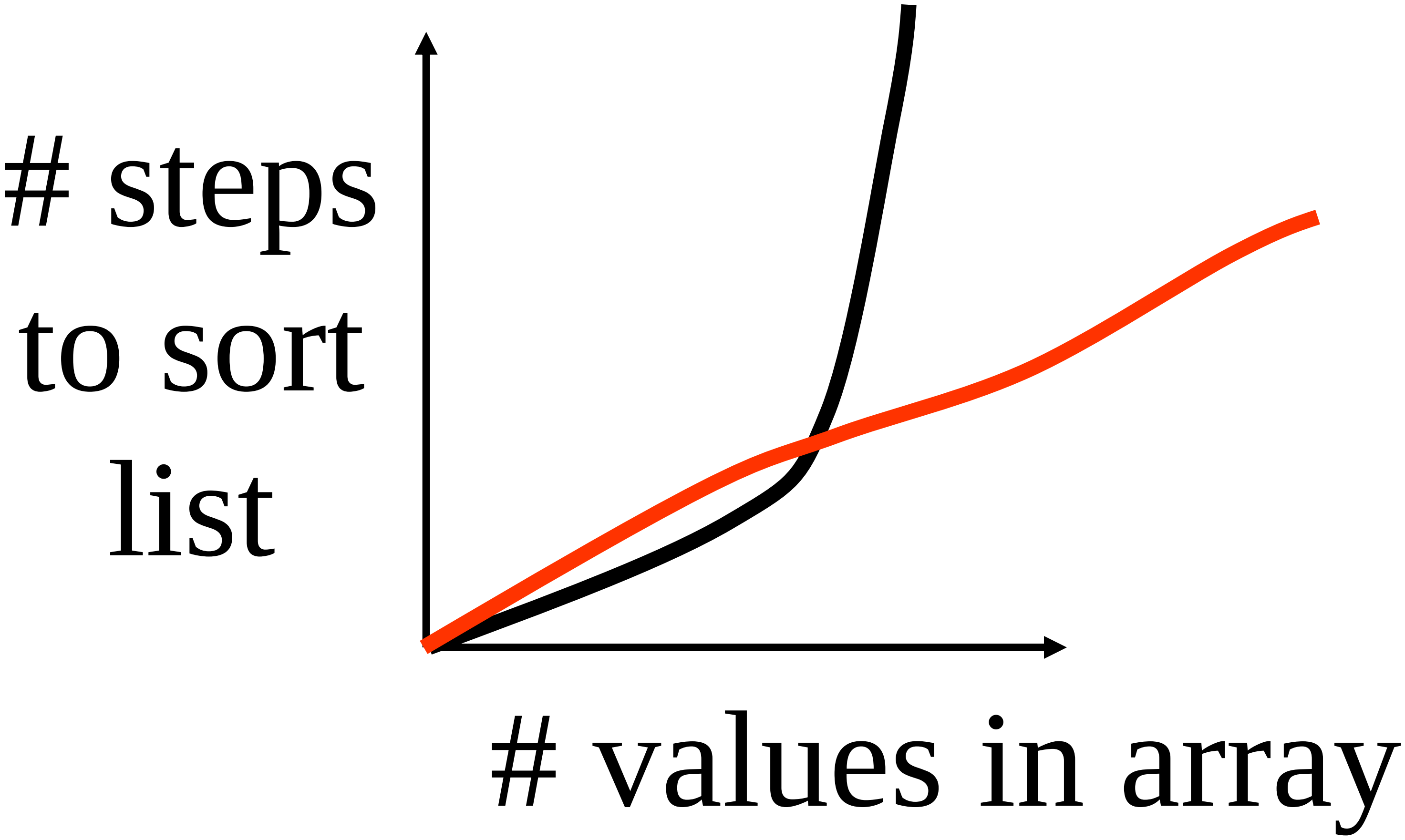
# Step Into Java: Recursion

Mr. Neat

Java

# Quadratic sorting algorithms are nice but...



**Need a new trick to do this…**

**recursion**

```
public void someMethod()
{
        // stuff
        anotherMethod();
        // more stuff
        // and more stuff
}
```

```
public void anotherMethod()
{
        // stuff
        oneMoreMethod();
        // more stuff
        // and more stuff
}
```

```
public void oneMoreMethod()
{
            // stuff
            // no more methods
            // more stuff
            // and more stuff
}
```

```
public void anotherMethod()
{
        // stuff
        oneMoreMethod();
        // more stuff
        // and more stuff
}
```

```
public void someMethod()
{
        // stuff
        anotherMethod();
        // more stuff
        // and more stuff
}
```

All done….

But what if…...

```
public void someMethod()
{
        // stuff
        someMethod();
        // more stuff
        // and more stuff
}
```

```
public void someMethod()
{
        // stuff
        someMethod();
        // more stuff
        // and more stuff
}
```

```
public void someMethod()
{
        // stuff
        someMethod();
        // more stuff
        // and more stuff
}
```

```
public void someMethod()
{
        // stuff
        someMethod();
        // more stuff
        // and more stuff
}
```

```
public void someMethod()
{
        // stuff
        someMethod();
        // more stuff
        // and more stuff
}
```

```
public void someMethod()
{
        // stuff
        someMethod();
        // more stuff
        // and more stuff
}
```

```
public void someMethod()
{
        // stuff
        someMethod();
        // more stuff
        // and more stuff
}
```

# Back to quiz question…..

Write a method that calculates n!

# Let’s walk through 4!

```
Sytem.out.print(fact(4));
```

```
public static int fact(int 4)
{
        if(1 = = 4)
        {
                return 1;
        }
        else
        {
                return 4*fact(3);
        }
}
```

```
public static int fact(int 4)
{
        if(1 = = 4)
        {
                return 1;
        }
        else
        {
                return 4*fact(3);
        }
}
```

```
public static int fact(int 3)
{
        if(1 = = 3)
        {
                return 1;
        }
        else
        {
                return 3*fact(2);
        }
}
```

```
fact(int 4)
{

        fact(int 3)
        {

        }


}
```

```
public static int fact(int 2)
{
        if(1 = = 2)
        {
                return 1;
        }
        else
        {
                return 2*fact(1);
        }
}
```

```
fact(int 4)
{

      fact(int 3)
      {
              fact(int 2)
              {

              }
      }

}
```

```
public static int fact(int 1)
{
          if(1 = = 1)
          {
                    return 1;
          }
          else
          {
                    return n*fact(n-1);
          }
}
```

```
fact(4)
{

        fact(3)
        {
                fact(2)
                {
                      fact(1);
                }

        }


}
```

```
fact(4)
{

        fact(3)
        {
                fact(2)
                {
                      return 2*fact(1);
                }

        }


}
```

```
fact(4)
{

        fact(3)
        {


               3*fact(2)


        }


}
```

```
fact(4)
{
     fact(3)
     {

          return 3*(2*fact(1));

     }

}
```

```
fact(4)
{

        return 4*fact(3);

}
```

```
fact(4)
{

        return 4*3*2*fact(1);

}
```

```
fact(4)
{
        return 4*3*2*1:
}
```

```
Sytem.out.print(fact(4));
```