USING THE MATHEMATICA IF STATEMENT

Last week we learned how to write Do, For and While loops in Mathematica. Today we have our introduction to conditional statements using the Mathematica "If" statement.

The If Statement

The If statement tests to see if a condition is true or false. We can begin learning about this command with the following simple example :

```
In[14]:= g[x_] := If[x > 0, Sqrt[x], 0]
         g[4]
Out[15]= 2

In[16]:= g[-4]
Out[16]= 0
```

I defined a function g[x]; my condition is if x > 0. If this condition is true, then the If statement will return the square root of x; if the condition is false (x < 0), the function will return zero.

We can use a similar form of If statement to integrate polynomials :

```
In[23]:= h[n_] := If[n ≠ -1, x^(n+1) / (n + 1), Log[x]]
         h[-3]
```

Out[24]= $-\frac{1}{2 x^2}$

```
In[26]:= h[-1]
Out[26]= Log[x]
```

These examples are not intended to substitute for the Mathematica integration function, but to show how an "If" statement works.

Sometimes we only want output when the statement is true; in such a case we can write an even simpler If statement of the form :

```
g[x_] := If[x > 0, x^(1 / 4)]
In[39]:=
```

This is a function that will return the fourth root of a positive number, such as

```
In[36]:= g[16]
Out[36]= 4
```

But for a negative value of x :

In[38]:=
```
g[-16]
```

We get no output. Try some examples of these in your notebooks.

Do Loops and the Permutation Tensor : Signature[{i, j, k}]

Let' s consider a more robust example. Recall that we encountered the "Signature" function when we studied the Levi - Civita permutation tensor. Signature[{i, j, k}] will return the value of the Levi - Civita tensor, as shown in the following examples :

In[28]:=
```
{Signature[{1, 2, 3}], Signature[{1, 3, 2}], Signature[{1, 1, 3}],
 Signature[{1, 2, 3, 4, 5}], Signature[{1, 3, 2, 4, 5}], Signature[{a, b, c, d}],
 Signature[{a, b, d, c}], Signature[{α, β, γ, δ}], Signature[{α, γ, β, δ}]}
```

Out[28]=
```
{1, -1, 0, 1, -1, 1, -1, 1, -1}
```

When we were studying the permutation tensor, we learned that there are 27 possible ways to permute the three coordinate indices in Cartesian 3 space. I can write a Do loop to print all 27 out :

```
Do[Print[Signature[{i, j, k}]], {i, 3}, {j, 3}, {k, 3}];
```

I have suppressed the output for obvious reasons, but if you input this statement you will get all 27 values. A slightly spiffier Print statement will list the values of i, j, k :

In[30]:=
```
Do[Print["ϵijk =  ", Signature[{i, j, k}], "   for i = ", i, "  j = ", j, "  k = ", k],
 {i, 3}, {j, 3}, {k, 3}]
```

```
∈ijk = 0   for i = 1  j = 1  k = 1
∈ijk = 0   for i = 1  j = 1  k = 2
∈ijk = 0   for i = 1  j = 1  k = 3
∈ijk = 0   for i = 1  j = 2  k = 1
∈ijk = 0   for i = 1  j = 2  k = 2
∈ijk = 1   for i = 1  j = 2  k = 3
∈ijk = 0   for i = 1  j = 3  k = 1
∈ijk = -1   for i = 1  j = 3  k = 2
∈ijk = 0   for i = 1  j = 3  k = 3
∈ijk = 0   for i = 2  j = 1  k = 1
∈ijk = 0   for i = 2  j = 1  k = 2
∈ijk = -1   for i = 2  j = 1  k = 3
∈ijk = 0   for i = 2  j = 2  k = 1
∈ijk = 0   for i = 2  j = 2  k = 2
∈ijk = 0   for i = 2  j = 2  k = 3
∈ijk = 1   for i = 2  j = 3  k = 1
∈ijk = 0   for i = 2  j = 3  k = 2
∈ijk = 0   for i = 2  j = 3  k = 3
∈ijk = 0   for i = 3  j = 1  k = 1
∈ijk = 1   for i = 3  j = 1  k = 2
∈ijk = 0   for i = 3  j = 1  k = 3
∈ijk = -1   for i = 3  j = 2  k = 1
∈ijk = 0   for i = 3  j = 2  k = 2
∈ijk = 0   for i = 3  j = 2  k = 3
∈ijk = 0   for i = 3  j = 3  k = 1
∈ijk = 0   for i = 3  j = 3  k = 2
∈ijk = 0   for i = 3  j = 3  k = 3
```

Now, the last examples were nice reviews of Do loops, but we are learning about If statements today.

I would like you to write an If statement that will produce output only if the value of the permutation tensor is non - zero.

For you final program today, read on the doc center about the PrimeQ and Fibonacci numbers, and write a short program that will test the first 100 Fibonacci numbers and print out values of any Fib. numbers that are prime.