

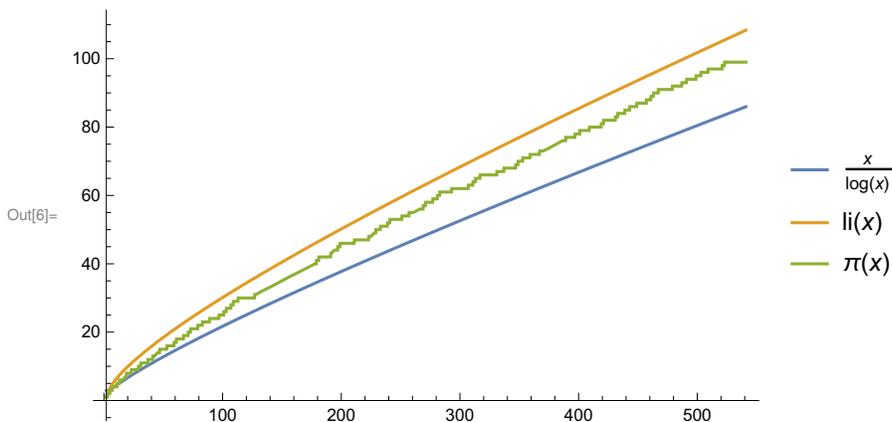
Winter 2019/20 — Algebra — TU Graz

Code

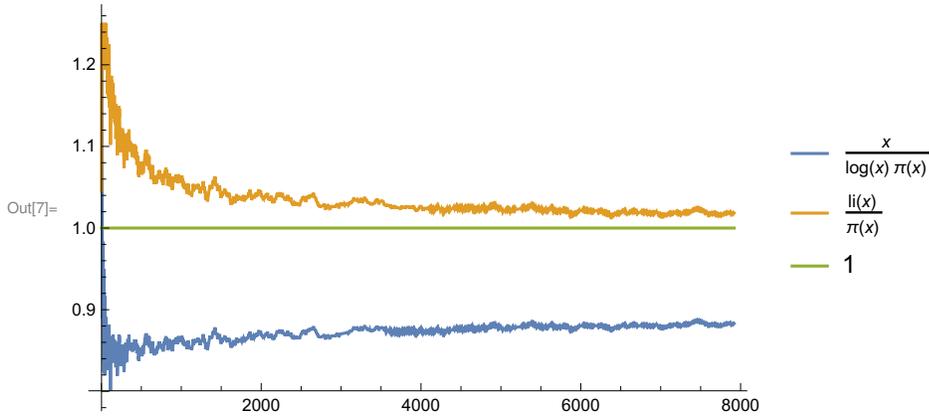
```
In[1]:= (* Das Ausführen von diesem Code benötigt einige Sekunden. *)
plot1 = Plot[{x/Log[x], LogIntegral[x], PrimePi[x]},
  {x, 2, Prime[100]}, PlotLegends -> "Expressions"];
plot2 = Plot[{(x/Log[x])/PrimePi[x], LogIntegral[x]/PrimePi[x], 1},
  {x, 2, Prime[1000]}, PlotLegends -> "Expressions"];
(* Ein Beweis dessen, dass die hier benutzte Formel wirklich die unten
  behauptete Anzahl berechnet, findet sich im Vorlesungsskriptum. *)
Pi[q_, n_] := Sum[MoebiusMu[d] q^n/d, {d, Divisors[n]}];
q = 2;
plot3 = DiscretePlot[{Log[Pi[q, n]], Log[q^n/n]},
  {n, 2, 100}, PlotLegends -> "Expressions"];
```

Visualisierungen zu Abschnitt § 4.3 im Vorlesungsskriptum

```
In[6]:= plot1 (* Plot der Anzahl aller Primzahlen ≤ x,
  nebst Approximationen  $\frac{x}{\log(x)}$ , sowie  $Li(x)$  *)
```



```
In[7]:= plot2 (* Plot der relativen Fehler der jeweiligen
Approximation an die Anzahl aller Primzahlen ≤ x. *)
```



```
In[8]:= plot3 (* Anzahl der normierten, irreduziblen Polynome P in F_q[X] mit deg P=n. *)
```

