We now build these f_k which give us the coordinates of the Jacobian near 0. Our differential dx/y which serves as f_1 needs to be rewritten in proper coordinates recursively. (This differential splits off from the rest of $H^0(\Omega^1_C)$ because all other differentials have a larger power of y in the bottom, and the group action by $\mathbb{G}_m$ acts on y.) We let $z = \frac{x}{y}$, and $w = \frac{1}{y}$. Then, we may rewrite the curve $\mathscr{C}$ at the origin as:

$$w = z^p + u_1 z^{p-1} w + \cdots + u_{p-2} z^2 w^{p-2} - (1 + u_1 + \cdots + u_{p-2}) z w^{p-1}.$$

Recursion is a fish by hand, here's my sage code. I personally did it to precision 5^4, but I will suppress the level of precision below for readablity.

```
sage: S.<u1,u2,u3> = ZZ[]
sage: L.<z> = LazyLaurentSeriesRing(S);
sage: w = L.series(lambda w,n: (z^5 + u1*z^4*w + u2*z^3*w^2 + u3*z^2*w
   ↪ ^3 - (1+u1+u2+u3)*z*w^4)[n], valuation=0)
sage: w.approximate_series(30)
z^5 + u1*z^9 + (u1^2 + u2)*z^13 + (u1^3 + 3*u1*u2 + u3)*z^17 + (u1^4 +
   ↪ 6*u1^2*u2 + 2*u2^2 + 4*u1*u3 - u1 - u2 - u3 - 1)*z^21 + (u1^5 +
   ↪ 10*u1^3*u2 + 10*u1*u2^2 + 10*u1^2*u3 - 5*u1^2 - 5*u1*u2 - 5*u1*u3
   ↪  + 5*u2*u3 - 5*u1)*z^25 + (u1^6 + 15*u1^4*u2 + 30*u1^2*u2^2 + 20*
   ↪ u1^3*u3 - 15*u1^3 - 15*u1^2*u2 + 5*u2^3 - 15*u1^2*u3 + 30*u1*u2*
   ↪ u3 - 15*u1^2 - 6*u1*u2 - 6*u2^2 - 6*u2*u3 + 3*u3^2 - 6*u2)*z^29 +
   ↪  O(z^30)
```

Since sage is currently unable to type change from Lazy to non-Lazy Rings, we have to stoop down to manually doing so, by copy pasting the output of the former computation of w above.

```
sage: S.<u1,u2,u3> = QQ[]
sage: L.<z> = LaurentSeriesRing(S);
sage: w = (output from above)
sage: y = 1/w
sage: x = y*z
sage: f = x.derivative()
sage: o = f/y
sage: o.integral()
```

```
-4*z - 4/5*u1*z^5 + (-4/9*u1^2 - 8/9*u2)*z^9 + (-4/13*u1^3 - 24/13*u1*
   ↪ u2 - 12/13*u3)*z^13 + (-4/17*u1^4 - 48/17*u1^2*u2 - 24/17*u2^2 -
   ↪ 48/17*u1*u3 + 16/17*u1 + 16/17*u2 + 16/17*u3 + 16/17)*z^17 +
   ↪ (-4/21*u1^5 - 80/21*u1^3*u2 - 40/7*u1*u2^2 - 40/7*u1^2*u3 +
   ↪ 80/21*u1^2 + 80/21*u1*u2 + 80/21*u1*u3 - 80/21*u2*u3 + 80/21*u1)*
   ↪ z^21 + (-4/25*u1^6 - 24/5*u1^4*u2 - 72/5*u1^2*u2^2 - 48/5*u1^3*u3
   ↪  + 48/5*u1^3 + 48/5*u1^2*u2 - 16/5*u2^3 + 48/5*u1^2*u3 - 96/5*u1*
   ↪ u2*u3 + 48/5*u1^2 + 24/5*u1*u2 + 24/5*u2^2 + 24/5*u2*u3 - 12/5*u3
   ↪ ^2 + 24/5*u2)*z^25 + (-4/29*u1^7 - 168/29*u1^5*u2 - 840/29*u1^3*
   ↪ u2^2 - 420/29*u1^4*u3 + 560/29*u1^4 + 560/29*u1^3*u2 - 560/29*u1*
   ↪ u2^3 + 560/29*u1^3*u3 - 1680/29*u1^2*u2*u3 + 560/29*u1^3 +
   ↪ 840/29*u1^2*u2 + 840/29*u1*u2^2 + 840/29*u1*u2*u3 - 420/29*u2^2*
   ↪ u3 - 420/29*u1*u3^2 + 840/29*u1*u2 + 168/29*u1*u3 + 168/29*u2*u3
   ↪ + 168/29*u3^2 + 168/29*u3)*z^29 + O(z^30)
```

Great, now we have written down our logarithm `o.integral()` $:= \int \frac{dx}{y}$, i.e., $\psi_{1,0}(x,y)$ in the notation of [?]. Recall that u_i is only well defined mod $(u_1, ..., u_{i-1})$. Modding out one by one by u_i as we increase i, we see that our logarithm is indeed universal, as it is isomorphic to the universal p-typical logarithm at each stage.