However, before long we will detect such a pair x_k, x_j whose difference has a common factor with n. Namely, suppose that k_0 has $h+1$ bits. Set $j = 2^{h+1} - 1$ and $k = j + (k_0 - j_0)$, in which case j is the largest $(h+1)$-bit integer and k is an $(h+2)$-bit integer such that $g.c.d.(x_k - x_j, n) > 1$. Notice that we have $k < 2^{h+2} = 4 \cdot 2^h \leq 4k_0$.

Example 2. Let us return to Example 1 but compare each x_k only with the particular x_j for which j is the largest integer $< k$ of the form $2^h - 1$. For $n = 91$, $f(x) = x^2 + 1$, $x_0 = 1$ we have $x_1 = 2$, $x_2 = 5$, $x_3 = 26$ as before, and $x_4 = 40$ (since $26^2 + 1 \equiv 40 \ mod \ 91$). Following the algorithm described above, we first find a factor of n when we compute $g.c.d.(x_4 - x_3, n) = g.c.d.(14, 91) = 7$.

Example 3. Factor 4087 using $f(x) = x^2 + x + 1$ and $x_0 = 2$.

Solution. Our computations proceed in the following order:

$$\begin{aligned}
&x_1 = f(2) = 7; \ g.c.d.(x_1 - x_0, n) = g.c.d.(7-2, 4087) = 1;\\
&x_2 = f(7) = 57; \ g.c.d.(x_2 - x_1, n) = g.c.d.(57-7, 4087) = 1;\\
&x_3 = f(57) = 3307; \ g.c.d.(x_3 - x_1, n) = g.c.d.(3307-7, 4087) = 1;\\
&x_4 \equiv f(3307) \equiv 2745 \ mod \ 4087; \ g.c.d.(x_4 - x_3, n)\\
&\quad = g.c.d.(2745 - 3307, 4087) = 1;\\
&x_5 \equiv f(2745) \equiv 1343 \ mod \ 4087; \ g.c.d.(x_5 - x_3, n)\\
&\quad = g.c.d.(1343 - 3307, 4087) = 1;\\
&x_6 \equiv f(1343) \equiv 2626 \ mod \ 4087; \ g.c.d.(x_6 - x_3, n)\\
&\quad = g.c.d.(2626 - 3307, 4087) = 1:\\
&x_7 \equiv f(2626) \equiv 3734 \ mod \ 4087; \ g.c.d.(x_7 - x_3, n)\\
&\quad = g.c.d.(3734 - 3307, 4087) = 61.
\end{aligned}$$

Thus, we obtain $4087 = 61 \cdot 67$, and we are done.

Proposition V.2.2. *Let n be an odd composite integer, and let r be a nontrivial divisor of n which is less than $\sqrt{n}$ (i.e., $r|n$, $1 < r < \sqrt{n}$; we suppose that we are trying to determine what r is). If a pair (f, x_0) consisting of a polynomial f with integer coefficients and an initial value x_0 is chosen which behaves like an average pair (f, x_0) in the sense of Proposition V.2.1 (with f a map from $\mathbf{Z}/r\mathbf{Z}$ to itself and x_0 an integer), then the rho method will reveal the factor r in $O(\sqrt[4]{n} \, log^3 n)$ bit operations with a high probability. More precisely, there exists a constant C such that for any positive real number λ the probability that the rho method fails to find a nontrivial factor of n in $C\sqrt{\lambda} \sqrt[4]{n} \, log^3 n$ bit operations is less than $e^{-\lambda}$.*

Proof. Let C_1 be a constant such that $g.c.d.(y - z, n)$ can be computed in $C_1 log^3 n$ bit operations whenever $y, z \leq n$ (see § I.3). Let C_2 be a constant such that the least nonnegative residue of $f(x)$ modulo n can be computed in $C_2 log^2 n$ bit operations whenever $x < n$ (see § I.1). If k_0 is the first index for which there exists $j_0 < k_0$ with $x_{k_0} \equiv x_{j_0} \ mod \ r$, then the rho