Universität des Saarlandes
Lehrstuhl für Softwaretechnik
Prof. Andreas Zeller, Stephan Neuhaus
Design of Secure Software Systems
Exercise 3

Total number of points obtainable: 17, plus 4 bonus points.

1 Mixing Functions

1.1 Statistical Distribution of 0 and 1 Bits

Assume that the probability of a 0 bit and the probability of a 1 bit in a bit string of length N are equal.

- **(1 point)** What is the expected number of 0 and 1 bits?
- **(1 point)** What is the standard deviation?

1.2 Experiments With Various Mixing Functions

Let n be a 32-bit unsigned integer. Compute a MD5 hash for all numbers between 0 and 9999999. Also compute $f(n) = 314159269n \bmod 2^{32}$. Compute the number of 0 and 1 bits in $f(n)$ and MD5 (n). Output the average number and standard deviation of 0 and 1 bits in $f(n)$ and MD5 (n).

- **(4 points)** Do the results support the conjecture that the bits in a MD5 hash are equally distributed? What about f? (You don't have to compute confidence levels; just a well-reasoned gut feeling will do.)
- **(4 points)** Based on these observations, which is the better mixing function, MD5 or f? Why?

Implementation Hint. Textbook formulas like this one for the standard deviation

$$\sigma = \sqrt{\left(\sum_{1\le k\le n} (x_k - \mu)^2\right) \Big/ (n-1)}$$

will suggest to you to gather all the values in a first pass and then to compute the average and standard deviation in a second pass. In this example, you'd have to allocate four arrays of size 10^7 each. Formulas like

$$\sigma = \sqrt{\left(n \sum_{1\le k\le n} x_k^2 - \left(\sum_{1\le k\le n} x_k\right)^2\right) \Big/ n(n-1)}$$

are no better, because they are fraught with numerical difficulties; for example, you may find yourself taking the square root of a negative number, because Cauchy's inequality doesn't hold for floating-point numbers. Actually, there is a better way to do it. If we want to compute the mean μ and standard deviation σ of a sample $x_1, \dots, x_n$, we use the recurrence formulas

$$\begin{aligned} M_1 &= x_1, \quad & M_k &= M_{k-1} + (x_k - M_{k-1})/k, \\ S_1 &= 0, \quad & S_k &= S_{k-1} + (x_k - M_{k-1})(x_k - M_k), \end{aligned}$$

for $2 \le k \le n$, and set $\mu = M_n$ and $\sigma = \sqrt{S_n/(n-1)}$. These formulas are numerically stable and should be used whenever running averages and standard deviations must be kept. (This trick is a bit too pretty not to be taught.)

2 Linear-Congruential Pseudo-Random Number Generators

2.1 Area of the Parallelepid

(3 points) Let $\langle X_n \rangle$ be a LCPRNG sequence defined by $X_{n+1} = (aX_n + c) \bmod m$. Prove that the area of the parallelepid defined by (X_0, X_1), (X_i, X_{i+1}), (X_j, X_{j+1}) is indeed

$$d(i,j) = \text{abs} \begin{vmatrix} X_i - X_0 & X_{i+1} - X_1 \\ X_j - X_0 & X_{j+1} - X_1 \end{vmatrix},$$

as claimed in the lecture.

Hint. What is the area of the parallelogram defined by two vectors?

2.2 Cracking a LCPRNG

(4 points) Find the parameters m, a, and c of the sequence beginning 1, 254, 130, 141, 225, 2, 28, 63, 120 . Use the method that was outlined in the lecture. You may want to write a computer program to automate most of the tasks.

3 Bonus Exercise: Modulo Operations in Your Head

Note. This exercise is not necessary to gain admission to the examination. All points you get are bonus points.

Sometimes it's possible to perform seemingly complicated modulo operations on paper, or even in your head. This could be useful if you happen to be stranded on a desert island and are suddenly struck with the urge to compute a modulus. (Seriously, though, these formulas can be useful when you're trying to verify a computation by a different route. Also, it's a really pretty trick.)

3.1 Modulo Base-Minus-One

One trick is a rule that some might know as the "division-by-9-rule". This rule says that if you have a number in base 10 and want to know its residue modulo 9, you can take the sum of the digits and take that modulo 9. For example, to compute $173562 \bmod 9$, compute $1 + 7 + 3 + 5 + 6 + 2 \equiv 24 \equiv 6 \pmod 9$, and indeed, $173562 = 19284 \cdot 9 + 6$.

- **(1 point)** Prove this rule.
- **(0.5 points)** Extend the rule from base 10 to arbitrary integer bases $b \geq 2$.
- **(0.5 points)** Explain how to use the rule to compute residues modulo 31 by hand.

3.2 Modulo Base-Plus-One

There is also a division-by-11 rule for residues modulo 11: Starting at the least significant digit, alternately subtract and add digits. For example, to compute $173562 \bmod 11$, compute $2 - 6 + 5 - 3 + 7 - 1 \equiv 4 \pmod{11}$, and indeed, $173562 = 15778 \cdot 11 + 4$.

- **(1 point)** Prove this rule.
- **(0.5 points)** Extend the rule from base 10 to arbitrary integer bases $b \geq 2$.
- **(0.5 points)** Explain how to use the rule to compute residues modulo 65537 by hand.