

#|

Copyright (C) 1994 by David Russinoff. All Rights Reserved.

You may copy and distribute verbatim copies of this Nqthm-1992 event script as you receive it, in any medium, including embedding it verbatim in derivative works, provided that you conspicuously and appropriately publish on each copy a valid copyright notice "Copyright (C) 1994 by David Russinoff. All Rights Reserved."

NO WARRANTY

David Russinoff PROVIDES ABSOLUTELY NO WARRANTY. THE EVENT SCRIPT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE SCRIPT IS WITH YOU. SHOULD THE SCRIPT PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT WILL David Russinoff BE LIABLE TO YOU FOR ANY DAMAGES, ANY LOST PROFITS, LOST MONIES, OR OTHER SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THIS SCRIPT (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY THIRD PARTIES), EVEN IF YOU HAVE ADVISED US OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY OTHER PARTY.

|#

; Work of David Russinoff.

EVENT: Start with the library "wilson" using the compiled version.

EVENT: For efficiency, compile those definitions not yet compiled.

DEFINITION:

squares( $n, p$ )

= **if**  $n \simeq 0$  **then** list(0)  
    **else** cons( $((n * n) \bmod p, \text{squares}(n - 1, p))$ ) **endif**

DEFINITION:

residue( $a, p$ ) =  $((\neg \text{divides}(p, a)) \wedge ((a \bmod p) \in \text{squares}(p, p)))$

THEOREM: all-squares-1

$((p \neq 0) \wedge (m \leq n)) \rightarrow (((m * m) \bmod p) \in \text{squares}(n, p))$

THEOREM: all-squares-2

$$((y * y) \bmod p) = (((y \bmod p) * (y \bmod p)) \bmod p)$$

THEOREM: all-squares

$$((p \neq 0) \wedge (x \notin \text{squares}(p, p))) \rightarrow (x \neq ((y * y) \bmod p))$$

THEOREM: euler-1-1

$$(\neg \text{divides}(2, p)) \rightarrow ((2 * (p \div 2)) = (p - 1))$$

THEOREM: euler-1-2

$$(\neg \text{divides}(2, p)) \rightarrow (\exp(i * i, p \div 2) = \exp(i, p - 1))$$

THEOREM: euler-1-3

$$((a \bmod p) = (b \bmod p)) \rightarrow ((\exp(a, c) \bmod p) = (\exp(b, c) \bmod p))$$

THEOREM: euler-1-4

$$\begin{aligned} &(\text{prime}(p) \wedge (\neg \text{divides}(2, p)) \wedge (\neg \text{divides}(p, i))) \\ \rightarrow &((\exp(i * i, p \div 2) \bmod p) = 1) \end{aligned}$$

THEOREM: euler-1-5

$$\begin{aligned} &(\text{prime}(p) \wedge (\neg \text{divides}(p, a)) \wedge ((a \bmod p) = ((i * i) \bmod p))) \\ \rightarrow &(\neg \text{divides}(p, i)) \end{aligned}$$

THEOREM: euler-1-6

$$\begin{aligned} &(\text{prime}(p) \\ &\wedge (\neg \text{divides}(2, p)) \\ &\wedge (\neg \text{divides}(p, a)) \\ &\wedge ((a \bmod p) = ((i * i) \bmod p))) \\ \rightarrow &((\exp(a, p \div 2) \bmod p) = 1) \end{aligned}$$

THEOREM: euler-1-7

$$\begin{aligned} &(\text{prime}(p) \\ &\wedge (\neg \text{divides}(2, p)) \\ &\wedge (\neg \text{divides}(p, a)) \\ &\wedge ((a \bmod p) \in \text{squares}(i, p))) \\ \rightarrow &((\exp(a, p \div 2) \bmod p) = 1) \end{aligned}$$

THEOREM: euler-1

$$\begin{aligned} &(\text{prime}(p) \wedge (\neg \text{divides}(2, p)) \wedge \text{residue}(a, p)) \\ \rightarrow &((\exp(a, p \div 2) \bmod p) = 1) \end{aligned}$$

DEFINITION: complement  $(j, a, p) = ((\text{inverse}(j, p) * a) \bmod p)$

EVENT: Enable inverse; name this event 'g0219'.

THEOREM: complement-works  
 $(\text{prime}(p) \wedge (\neg \text{divides}(p, j)))$   
 $\rightarrow ((j * \text{complement}(j, a, p)) \bmod p) = (a \bmod p)$

THEOREM: bounded-complement  
 $(p \neq 0) \rightarrow (\text{complement}(j, a, p) < p)$

EVENT: Enable complement; name this event ‘complement-off’.

THEOREM: non-zero-complement  
 $(\text{prime}(p) \wedge (\neg \text{divides}(p, j)) \wedge (\neg \text{divides}(p, a)))$   
 $\rightarrow (\text{complement}(j, a, p) \neq 0)$

THEOREM: complement-is-unique  
 $(\text{prime}(p) \wedge (\neg \text{divides}(p, a)) \wedge ((j * x) \bmod p) = (a \bmod p))$   
 $\rightarrow (\text{complement}(j, a, p) = (x \bmod p))$

EVENT: Enable squares; name this event ‘squares-off’.

THEOREM: no-self-complement  
 $(\text{prime}(p)$   
 $\wedge (\neg \text{divides}(p, j))$   
 $\wedge (\neg \text{divides}(p, a))$   
 $\wedge (\neg \text{residue}(a, p)))$   
 $\rightarrow (j \neq \text{complement}(j, a, p))$

THEOREM: complement-of-complement  
 $(\text{prime}(p) \wedge (\neg \text{divides}(p, j)) \wedge (\neg \text{divides}(p, a)))$   
 $\rightarrow (\text{complement}(\text{complement}(j, a, p), a, p) = (j \bmod p))$

DEFINITION:  
 $\text{comp-list}(i, a, p)$   
 $=$  **if**  $i \simeq 0$  **then nil**  
       **elseif**  $i \in \text{comp-list}(i - 1, a, p)$  **then**  $\text{comp-list}(i - 1, a, p)$   
       **else**  $\text{cons}(i, \text{cons}(\text{complement}(i, a, p), \text{comp-list}(i - 1, a, p)))$  **endif**

THEOREM: all-non-zero-comp-list  
 $(\text{prime}(p) \wedge (i < p) \wedge (\neg \text{divides}(p, a)))$   
 $\rightarrow \text{all-non-zero}(\text{comp-list}(i, a, p))$

THEOREM: bounded-comp-list  
 $(i < p) \rightarrow \text{all-lesseqp}(\text{comp-list}(i, a, p), p - 1)$

THEOREM: subsetp-positives-comp-list  
 $\text{subsetp}(\text{positives}(n), \text{comp-list}(n, a, p))$

THEOREM: comp-list-closed-1

$$\begin{aligned}
& (\text{prime } (p)) \\
& \wedge \quad (i \neq 0) \\
& \wedge \quad (i < p) \\
& \wedge \quad (\neg \text{divides } (p, a)) \\
& \wedge \quad (j \in \text{comp-list } (i, a, p))) \\
& \rightarrow \quad (\text{complement } (j, a, p) \in \text{comp-list } (i, a, p))
\end{aligned}$$

THEOREM: comp-list-closed-2

$$\begin{aligned}
& (\text{prime } (p)) \\
& \wedge \quad (i \neq 0) \\
& \wedge \quad (j \neq 0) \\
& \wedge \quad (i < p) \\
& \wedge \quad (j < p) \\
& \wedge \quad (\neg \text{divides } (p, a)) \\
& \wedge \quad (\text{complement } (j, a, p) \in \text{comp-list } (i, a, p))) \\
& \rightarrow \quad (j \in \text{comp-list } (i, a, p))
\end{aligned}$$

THEOREM: all-distinct-comp-list-1

$$\begin{aligned}
& (\text{prime } (p)) \\
& \wedge \quad (i < p) \\
& \wedge \quad (\neg \text{divides } (p, a)) \\
& \wedge \quad (\neg \text{residue } (a, p)) \\
& \wedge \quad \text{all-distinct } (\text{comp-list } (i - 1, a, p))) \\
& \rightarrow \quad \text{all-distinct } (\text{comp-list } (i, a, p))
\end{aligned}$$

THEOREM: all-distinct-comp-list

$$\begin{aligned}
& (\text{prime } (p) \wedge (i < p) \wedge (\neg \text{divides } (p, a)) \wedge (\neg \text{residue } (a, p))) \\
& \rightarrow \quad \text{all-distinct } (\text{comp-list } (i, a, p))
\end{aligned}$$

THEOREM: perm-positives-comp-list

$$\begin{aligned}
& (\text{prime } (p) \wedge (\neg \text{divides } (p, a)) \wedge (\neg \text{residue } (a, p))) \\
& \rightarrow \quad \text{perm } (\text{positives } (p - 1), \text{comp-list } (p - 1, a, p))
\end{aligned}$$

THEOREM: comp-list-fact

$$\begin{aligned}
& (\text{prime } (p) \wedge (\neg \text{divides } (p, a)) \wedge (\neg \text{residue } (a, p))) \\
& \rightarrow \quad (\text{times-list } (\text{comp-list } (p - 1, a, p)) = \text{fact } (p - 1))
\end{aligned}$$

THEOREM: times-mod-4

$$\begin{aligned}
& (((i * j) \bmod p) = (a \bmod p)) \\
& \rightarrow \quad (((i * (j * k)) \bmod p) = ((a * (k \bmod p)) \bmod p))
\end{aligned}$$

THEOREM: times-comp-list-1

$$\begin{aligned}
& (((i * \text{complement } (i, a, p)) \bmod p) = (a \bmod p)) \\
& \wedge \quad (i \neq 0)
\end{aligned}$$

$$\begin{aligned}
& \wedge (i \notin \text{comp-list}(i-1, a, p)) \\
\rightarrow & ((\text{times-list}(\text{comp-list}(i, a, p)) \bmod p) \\
& = ((a * (\text{times-list}(\text{comp-list}(i-1, a, p)) \bmod p)) \bmod p))
\end{aligned}$$

THEOREM: times-comp-list-2

$$\begin{aligned}
& (\text{prime}(p) \wedge (\neg \text{divides}(p, i)) \wedge (i \notin \text{comp-list}(i-1, a, p))) \\
\rightarrow & ((\text{times-list}(\text{comp-list}(i, a, p)) \bmod p) \\
& = ((a * (\text{times-list}(\text{comp-list}(i-1, a, p)) \bmod p)) \bmod p))
\end{aligned}$$

THEOREM: quotient-plus-1

$$\begin{aligned}
& ((n \neq 0) \wedge (x \in \mathbf{N}) \wedge (y = (x + n))) \\
\rightarrow & ((y \div n) = (1 + (x \div n)))
\end{aligned}$$

THEOREM: times-comp-list-3

$$\begin{aligned}
& ((i \neq 0) \wedge (i \notin \text{comp-list}(i-1, a, p))) \\
\rightarrow & ((\text{length}(\text{comp-list}(i, a, p)) \div 2) \\
& = (1 + (\text{length}(\text{comp-list}(i-1, a, p)) \div 2)))
\end{aligned}$$

THEOREM: times-comp-list-4

$$\begin{aligned}
& (\text{prime}(p) \\
& \wedge (i \neq 0) \\
& \wedge (i < p) \\
& \wedge ((\text{times-list}(\text{comp-list}(i-1, a, p)) \bmod p) \\
& = (\exp(a, \text{length}(\text{comp-list}(i-1, a, p)) \div 2) \bmod p))) \\
\rightarrow & ((\text{times-list}(\text{comp-list}(i, a, p)) \bmod p) \\
& = (\exp(a, \text{length}(\text{comp-list}(i, a, p)) \div 2) \bmod p))
\end{aligned}$$

THEOREM: times-comp-list-5

$$\begin{aligned}
& (i \simeq 0) \\
\rightarrow & ((\text{times-list}(\text{comp-list}(i, a, p)) \bmod p) \\
& = (\exp(a, \text{length}(\text{comp-list}(i, a, p)) \div 2) \bmod p))
\end{aligned}$$

THEOREM: times-comp-list

$$\begin{aligned}
& (\text{prime}(p) \wedge (i < p)) \\
\rightarrow & ((\text{times-list}(\text{comp-list}(i, a, p)) \bmod p) \\
& = (\exp(a, \text{length}(\text{comp-list}(i, a, p)) \div 2) \bmod p))
\end{aligned}$$

THEOREM: sub1-length-delete

$$(x \in b) \rightarrow (\text{length}(\text{delete}(x, b)) = (\text{length}(b) - 1))$$

THEOREM: equal-length-perm

$$\text{perm}(a, b) \rightarrow (\text{length}(a) = \text{length}(b))$$

THEOREM: length-positives

$$\text{length}(\text{positives}(n)) = \text{fix}(n)$$

THEOREM: euler-2-1

$$\begin{aligned} & (\text{prime}(p) \wedge (\neg \text{divides}(p, a)) \wedge (\neg \text{residue}(a, p))) \\ \rightarrow & ((\exp(a, \text{length}(\text{comp-list}(p-1, a, p)) \div 2) \bmod p) = (p-1)) \end{aligned}$$

THEOREM: euler-2-2

$$\begin{aligned} & (\text{prime}(p) \wedge (\neg \text{divides}(p, a)) \wedge (\neg \text{residue}(a, p))) \\ \rightarrow & (\text{length}(\text{comp-list}(p-1, a, p)) = (p-1)) \end{aligned}$$

THEOREM: euler-2-3

$$(p \neq 0) \rightarrow (\text{divides}(2, p) = (\neg \text{divides}(2, p-1)))$$

THEOREM: euler-2-4

$$(\neg \text{divides}(2, p)) \rightarrow (((p-1) \div 2) = (p \div 2))$$

THEOREM: euler-2

$$\begin{aligned} & (\text{prime}(p) \\ & \wedge (\neg \text{divides}(2, p)) \\ & \wedge (\neg \text{divides}(p, a)) \\ & \wedge (\neg \text{residue}(a, p))) \\ \rightarrow & ((\exp(a, p \div 2) \bmod p) = (p-1)) \end{aligned}$$

DEFINITION:

$$\begin{aligned} & \text{res1}(n, a, p) \\ = & \text{if } n \simeq 0 \text{ then } \mathbf{t} \\ & \quad \text{elseif } (p \div 2) < ((a * n) \bmod p) \text{ then } \neg \text{res1}(n-1, a, p) \\ & \quad \text{else } \text{res1}(n-1, a, p) \text{ endif} \end{aligned}$$

DEFINITION:  $\text{reflect}(x, p) = (p - x)$

DEFINITION:

$$\begin{aligned} & \text{reflect-list}(n, a, p) \\ = & \text{if } n \simeq 0 \text{ then } \mathbf{nil} \\ & \quad \text{elseif } (p \div 2) < ((a * n) \bmod p) \\ & \quad \text{then } \text{cons}(\text{reflect}((a * n) \bmod p, p), \text{reflect-list}(n-1, a, p)) \\ & \quad \text{else } \text{cons}((a * n) \bmod p, \text{reflect-list}(n-1, a, p)) \text{ endif} \end{aligned}$$

THEOREM: diff-mod-1

$$(b \leq a) \rightarrow (((a - (b \bmod p)) \bmod p) = ((a - b) \bmod p))$$

THEOREM: rem-diff-times

$$\begin{aligned} & ((x < p) \wedge (x \neq 0) \wedge (b \neq 0)) \\ \rightarrow & (((b * p) - x) \bmod p) = (p - x) \end{aligned}$$

THEOREM: reflect-commutes-with-times-1

$$\begin{aligned} & (y \leq p) \\ \rightarrow & (((\text{reflect}(y, p) * x) \bmod p) = (\text{reflect}((y * x) \bmod p, p) \bmod p)) \end{aligned}$$

THEOREM: reflect-commutes-with-times-2

$$(y \leq p) \rightarrow (((x * \text{reflect}(y, p)) \bmod p) = (\text{reflect}((x * y) \bmod p, p) \bmod p))$$

THEOREM: times-exp-fact

$$(n \neq 0) \rightarrow (((a * n) * (\exp(a, n - 1) * \text{fact}(n - 1))) \bmod p) = ((\exp(a, n) * \text{fact}(n)) \bmod p)$$

THEOREM: rem-reflect-list-1

$$\begin{aligned} & ((p \neq 0) \\ & \wedge (n \neq 0) \\ & \wedge ((p \div 2) \nless ((a * n) \bmod p)) \\ & \wedge (((\text{times-list}(\text{reflect-list}(n - 1, a, p)) \bmod p) \\ & \quad = ((\exp(a, n - 1) * \text{fact}(n - 1)) \bmod p))) \\ & \rightarrow ((\text{times-list}(\text{reflect-list}(n, a, p)) \bmod p) \\ & \quad = ((\exp(a, n) * \text{fact}(n)) \bmod p)) \end{aligned}$$

THEOREM: rem-reflect-list-2

$$\begin{aligned} & ((p \neq 0) \\ & \wedge (n \neq 0) \\ & \wedge ((p \div 2) < ((a * n) \bmod p)) \\ & \wedge (((\text{times-list}(\text{reflect-list}(n - 1, a, p)) \bmod p) \\ & \quad = ((\exp(a, n - 1) * \text{fact}(n - 1)) \bmod p))) \\ & \rightarrow ((\text{times-list}(\text{reflect-list}(n, a, p)) \bmod p) \\ & \quad = (\text{reflect}((\exp(a, n) * \text{fact}(n)) \bmod p, p) \bmod p)) \end{aligned}$$

THEOREM: rem-reflect-list-3

$$\begin{aligned} & ((p \neq 0) \\ & \wedge (n \neq 0) \\ & \wedge ((p \div 2) \nless ((a * n) \bmod p)) \\ & \wedge (((\text{times-list}(\text{reflect-list}(n - 1, a, p)) \bmod p) \\ & \quad = (\text{reflect}((\exp(a, n - 1) * \text{fact}(n - 1)) \bmod p, p) \bmod p))) \\ & \rightarrow ((\text{times-list}(\text{reflect-list}(n, a, p)) \bmod p) \\ & \quad = (\text{reflect}((\exp(a, n) * \text{fact}(n)) \bmod p, p) \bmod p)) \end{aligned}$$

THEOREM: double-reflect

$$(a \leq p) \rightarrow ((\text{reflect}(\text{reflect}(a, p) \bmod p, p) \bmod p) = (a \bmod p))$$

THEOREM: rem-reflect-list-4

$$\begin{aligned} & ((p \neq 0) \\ & \wedge (n \neq 0) \\ & \wedge ((p \div 2) < ((a * n) \bmod p)) \\ & \wedge (((\text{times-list}(\text{reflect-list}(n - 1, a, p)) \bmod p) \\ & \quad = (\text{reflect}((\exp(a, n - 1) * \text{fact}(n - 1)) \bmod p, p) \bmod p))) \\ & \rightarrow ((\text{times-list}(\text{reflect-list}(n, a, p)) \bmod p) \\ & \quad = ((\exp(a, n) * \text{fact}(n)) \bmod p)) \end{aligned}$$

THEOREM: rem-reflect-list-base-case

$$\begin{aligned} & (n \simeq 0) \\ \rightarrow & ((\text{times-list } (\text{reflect-list } (n, a, p)) \text{ mod } p) \\ & = ((\text{exp } (a, n) * \text{fact } (n)) \text{ mod } p)) \end{aligned}$$

THEOREM: rem-reflect-list

$$\begin{aligned} & (p \neq 0) \\ \rightarrow & ((\text{times-list } (\text{reflect-list } (n, a, p)) \text{ mod } p) \\ & = \text{ if } \text{res1 } (n, a, p) \text{ then } (\text{exp } (a, n) * \text{fact } (n)) \text{ mod } p \\ & \quad \text{else } \text{reflect } ((\text{exp } (a, n) * \text{fact } (n)) \text{ mod } p, p) \text{ mod } p \text{ endif}) \end{aligned}$$

THEOREM: length-reflect-list

$$\text{length } (\text{reflect-list } (n, a, p)) = \text{fix } (n)$$

THEOREM: all-lesseqp-reflect-list-1

$$((p \div 2) < x) \rightarrow ((p \div 2) \not< \text{reflect } (x, p))$$

THEOREM: all-lesseqp-reflect-list

$$\text{all-lesseqp } (\text{reflect-list } (n, a, p), p \div 2)$$

THEOREM: all-non-zerop-reflect-list

$$\begin{aligned} & (\text{prime } (p) \wedge (\neg \text{divides } (p, a)) \wedge (b < p)) \\ \rightarrow & \text{all-non-zerop } (\text{reflect-list } (b, a, p)) \end{aligned}$$

THEOREM: all-distinct-reflect-list-1

$$\begin{aligned} & (\text{prime } (p) \wedge (j < i) \wedge (i < p) \wedge (\neg \text{divides } (p, a))) \\ \rightarrow & (((a * i) \text{ mod } p) \neq ((a * j) \text{ mod } p)) \end{aligned}$$

THEOREM: all-distinct-reflect-list-2

$$\begin{aligned} & ((x \in \mathbf{N}) \wedge (y \in \mathbf{N}) \wedge (x < p) \wedge (y < p)) \\ \rightarrow & (((p - x) = (p - y)) = (x = y)) \end{aligned}$$

THEOREM: numberp-remainder

$$(a \text{ mod } p) \in \mathbf{N}$$

THEOREM: all-distinct-reflect-list-3

$$\begin{aligned} & (\text{prime } (p) \wedge (j < i) \wedge (i < p) \wedge (\neg \text{divides } (p, a))) \\ \rightarrow & (\text{reflect } ((a * i) \text{ mod } p, p) \neq \text{reflect } ((a * j) \text{ mod } p, p)) \end{aligned}$$

THEOREM: plus-mod-1

$$(((x \text{ mod } p) + y) \text{ mod } p) = ((x + y) \text{ mod } p)$$

THEOREM: plus-mod-2

$$((y + (x \text{ mod } p)) \text{ mod } p) = ((x + y) \text{ mod } p)$$

THEOREM: all-distinct-reflect-list-4

$$((x = (p - y)) \wedge (y < p)) \rightarrow (((x + y) \text{ mod } p) = 0)$$



THEOREM: all-distinct-reflect-list-5

$$\begin{aligned} & (((a * i) \bmod p) = (p - ((a * j) \bmod p))) \wedge (p \neq 0) \\ \rightarrow & (((a * (i + j)) \bmod p) = 0) \end{aligned}$$

THEOREM: all-distinct-reflect-list-6

$$\begin{aligned} & ((i \leq (p \div 2)) \wedge (j < i)) \\ \rightarrow & (((i + j) \neq 0) \wedge ((i + j) < p)) \end{aligned}$$

THEOREM: all-distinct-reflect-list-7

$$\begin{aligned} & (\text{prime}(p) \wedge (\neg \text{divides}(p, a)) \wedge (i \leq (p \div 2)) \wedge (j < i)) \\ \rightarrow & (((a * i) \bmod p) \neq \text{reflect}((a * j) \bmod p, p)) \end{aligned}$$

THEOREM: all-distinct-reflect-list-8

$$\begin{aligned} & (\text{prime}(p) \wedge (\neg \text{divides}(p, a)) \wedge (i \leq (p \div 2)) \wedge (j < i)) \\ \rightarrow & (\text{reflect}((a * i) \bmod p, p) \neq ((a * j) \bmod p)) \end{aligned}$$

THEOREM: all-distinct-reflect-list-9

$$\begin{aligned} & (\text{prime}(p) \\ & \wedge (\neg \text{divides}(2, p)) \\ & \wedge (\neg \text{divides}(p, a)) \\ & \wedge (i \leq (p \div 2)) \\ & \wedge (j < i)) \\ \rightarrow & (((a * i) \bmod p) \notin \text{reflect-list}(j, a, p)) \end{aligned}$$

THEOREM: all-distinct-reflect-list-10

$$\begin{aligned} & (\text{prime}(p) \\ & \wedge (\neg \text{divides}(2, p)) \\ & \wedge (\neg \text{divides}(p, a)) \\ & \wedge (i \leq (p \div 2)) \\ & \wedge (j < i)) \\ \rightarrow & (\text{reflect}((a * i) \bmod p, p) \notin \text{reflect-list}(j, a, p)) \end{aligned}$$

THEOREM: all-distinct-reflect-list

$$\begin{aligned} & (\text{prime}(p) \\ & \wedge (\neg \text{divides}(2, p)) \\ & \wedge (\neg \text{divides}(p, a)) \\ & \wedge (i \leq (p \div 2))) \\ \rightarrow & \text{all-distinct}(\text{reflect-list}(i, a, p)) \end{aligned}$$

THEOREM: times-reflect-list

$$\begin{aligned} & (\text{prime}(p) \wedge (\neg \text{divides}(2, p)) \wedge (\neg \text{divides}(p, a))) \\ \rightarrow & (\text{times-list}(\text{reflect-list}(p \div 2, a, p)) = \text{fact}(p \div 2)) \end{aligned}$$

THEOREM: plus-x-x-even

$$((x + x) \bmod 2) = 0$$

THEOREM: res1-rem-1-1

$$((x \neq 0) \wedge (\neg \text{divides}(2, p))) \rightarrow ((p - x) \bmod p \neq x)$$

THEOREM: res1-rem-1

$$\begin{aligned} & (\text{prime}(p) \\ & \wedge (\neg \text{divides}(2, p)) \\ & \wedge (\neg \text{divides}(p, a)) \\ & \wedge \text{res1}(p \div 2, a, p)) \\ & \rightarrow ((\exp(a, p \div 2) \bmod p) = 1) \end{aligned}$$

THEOREM: remainder-lessp

$$(a < p) \rightarrow ((a \bmod p) = \text{fix}(a))$$

THEOREM: res1-rem-2

$$\begin{aligned} & (\text{prime}(p) \\ & \wedge (\neg \text{divides}(2, p)) \\ & \wedge (\neg \text{divides}(p, a)) \\ & \wedge (\neg \text{res1}(p \div 2, a, p))) \\ & \rightarrow ((\exp(a, p \div 2) \bmod p) \neq 1) \end{aligned}$$

THEOREM: two-even

$$(\neg \text{divides}(2, p)) \rightarrow ((p - 1) \neq 1)$$

THEOREM: gauss-lemma

$$\begin{aligned} & (\text{prime}(p) \wedge (\neg \text{divides}(p, a)) \wedge (\neg \text{divides}(2, p))) \\ & \rightarrow (\text{res1}(p \div 2, a, p) = \text{residue}(a, p)) \end{aligned}$$

DEFINITION:

$$\begin{aligned} & \text{plus-list}(l) \\ & = \text{if } l \simeq \text{nil} \text{ then } 0 \\ & \quad \text{else } \text{car}(l) + \text{plus-list}(\text{cdr}(l)) \text{ endif} \end{aligned}$$

DEFINITION:

$$\begin{aligned} & \text{quot-list}(n, a, p) \\ & = \text{if } n \simeq 0 \text{ then nil} \\ & \quad \text{else } \text{cons}((a * n) \div p, \text{quot-list}(n - 1, a, p)) \text{ endif} \end{aligned}$$

DEFINITION:

$$\begin{aligned} & \text{rem-list}(n, a, p) \\ & = \text{if } n \simeq 0 \text{ then nil} \\ & \quad \text{else } \text{cons}((a * n) \bmod p, \text{rem-list}(n - 1, a, p)) \text{ endif} \end{aligned}$$

THEOREM: rem-quot-list

$$\begin{aligned} & (a * \text{plus-list}(\text{positives}(n))) \\ & = ((p * \text{plus-list}(\text{quot-list}(n, a, p))) + \text{plus-list}(\text{rem-list}(n, a, p))) \end{aligned}$$

DEFINITION:

$\text{even3}(x)$   
 $=$  **if**  $x \simeq 0$  **then** **t**  
       **else**  $\neg \text{even3}(x - 1)$  **endif**

THEOREM: even3-plus

$\text{even3}(a + b) = (\text{even3}(a) = \text{even3}(b))$

THEOREM: even3-diff

$(x \leq p) \rightarrow (\text{even3}(p - x) = (\text{even3}(p) = \text{even3}(x)))$

THEOREM: even3-times

$\text{even3}(a * b) = (\text{even3}(a) \vee \text{even3}(b))$

THEOREM: even3-rem

$(\neg \text{even3}(p)) \rightarrow (\text{even3}(p - (x \bmod p)) = (\neg \text{even3}(x \bmod p)))$

THEOREM: even3-rem-reflect

$(\neg \text{even3}(p))$   
 $\rightarrow$   $(\text{res1}(n, a, p)$   
        $=$   $(\text{even3}(\text{plus-list}(\text{rem-list}(n, a, p)))$   
        $\leftrightarrow$   $\text{even3}(\text{plus-list}(\text{reflect-list}(n, a, p))))$

THEOREM: perm-plus-list-1

$(x \in m) \rightarrow ((x + \text{plus-list}(\text{delete}(x, m))) = \text{plus-list}(m))$

THEOREM: perm-plus-list

$\text{perm}(l, m) \rightarrow (\text{plus-list}(l) = \text{plus-list}(m))$

THEOREM: even3-even

$\text{divides}(2, p) = \text{even3}(p)$

THEOREM: plus-reflect-list

$(\text{prime}(p) \wedge (\neg \text{divides}(p, a)) \wedge (\neg \text{even3}(p)))$   
 $\rightarrow$   $(\text{plus-list}(\text{reflect-list}(p \div 2, a, p)))$   
        $=$   $\text{plus-list}(\text{positives}(p \div 2))$

THEOREM: equals-have-same-parity

$(x = y) \rightarrow (\text{even3}(x) = \text{even3}(y))$

THEOREM: res1-quot-list

$(\text{prime}(p) \wedge (\neg \text{even3}(p)) \wedge (\neg \text{even3}(a)) \wedge (\neg \text{divides}(p, a)))$   
 $\rightarrow$   $(\text{res1}(p \div 2, a, p) = \text{even3}(\text{plus-list}(\text{quot-list}(p \div 2, a, p))))$

DEFINITION:

$\text{wins1}(x, l)$   
 $=$  **if**  $l \simeq \text{nil}$  **then** **0**  
       **elseif**  $\text{car}(l) < x$  **then**  $1 + \text{wins1}(x, \text{cdr}(l))$   
       **else**  $\text{wins1}(x, \text{cdr}(l))$  **endif**

DEFINITION:

wins( $k, l$ )  
= **if**  $k \simeq \mathbf{nil}$  **then** 0  
  **else** wins1(car( $k$ ),  $l$ ) + wins(cdr( $k$ ),  $l$ ) **endif**

DEFINITION:

losses1( $x, l$ )  
= **if**  $l \simeq \mathbf{nil}$  **then** 0  
  **elseif**  $x < \text{car}(l)$  **then** 1 + losses1( $x$ , cdr( $l$ ))  
  **else** losses1( $x$ , cdr( $l$ )) **endif**

DEFINITION:

losses( $k, l$ )  
= **if**  $k \simeq \mathbf{nil}$  **then** 0  
  **else** losses1(car( $k$ ),  $l$ ) + losses(cdr( $k$ ),  $l$ ) **endif**

THEOREM: win-some-lose-some-1

(( $x \notin l$ )  $\wedge$  all-non-zero( $l$ ))  
→ ((losses1( $x, l$ ) + wins1( $x, l$ )) = length( $l$ ))

THEOREM: win-some-lose-some-2

((intersect( $l, m$ )  $\simeq \mathbf{nil}$ )  $\wedge$  all-non-zero( $l$ )  $\wedge$  all-non-zero( $m$ ))  
→ ((wins( $l, m$ ) + losses( $l, m$ )) = (length( $l$ ) \* length( $m$ )))

THEOREM: equal-losses-wins

losses( $l, m$ ) = wins( $m, l$ )

THEOREM: a-winner-every-time

((intersect( $l, m$ )  $\simeq \mathbf{nil}$ )  $\wedge$  all-non-zero( $l$ )  $\wedge$  all-non-zero( $m$ ))  
→ ((wins( $l, m$ ) + wins( $m, l$ )) = (length( $l$ ) \* length( $m$ )))

DEFINITION:

mults( $n, p$ )  
= **if**  $n \simeq 0$  **then**  $\mathbf{nil}$   
  **else** cons( $n * p$ , mults( $n - 1, p$ )) **endif**

THEOREM: length-mults

length(mults( $n, p$ )) = fix( $n$ )

THEOREM: leq-n-wins1

(( $n * p < a$ ) → ( $n \leq \text{wins1}(a, \text{mults}(n, p))$ ))

THEOREM: monotone-wins1

( $n \leq m$ ) → ( $\text{wins1}(a, \text{mults}(n, p)) \leq \text{wins1}(a, \text{mults}(m, p))$ )

DEFINITION:

quot-quot-induction( $a, b, c, d$ )

= **if**  $b \simeq 0$  **then** **t**  
     **elseif**  $d \simeq 0$  **then** **t**  
     **elseif**  $a < d$  **then** **t**  
     **elseif**  $c < b$  **then** **t**  
     **else** quot-quot-induction( $a - d, b, c - b, d$ ) **endif**

THEOREM: leq-times-quot

$((b \neq 0) \wedge ((a * b) \leq (c * d))) \rightarrow ((a \div d) \leq (c \div b))$

THEOREM: leq-quot-times

$((p \div 2) * q) \div p \leq (q \div 2)$

DEFINITION:

monotone-quot-induction( $i, j, p$ )

= **if**  $p \simeq 0$  **then** **t**  
     **elseif**  $i < p$  **then** **t**  
     **elseif**  $j < p$  **then** **t**  
     **else** monotone-quot-induction( $i - p, j - p, p$ ) **endif**

THEOREM: monotone-quot

$(j \leq i) \rightarrow ((j \div p) \leq (i \div p))$

THEOREM: leq-quot-times-2

$(j \leq (p \div 2)) \rightarrow (((j * q) \div p) \leq (q \div 2))$

THEOREM: leq-quot-wins1-1

$(\neg \text{divides}(p, x)) \rightarrow (((x \div p) * p) < x)$

THEOREM: leq-quot-wins1-2

$(\text{prime}(p) \wedge (\neg \text{divides}(p, q)) \wedge (q \neq 0) \wedge (j \neq 0) \wedge (j < p))$   
 $\rightarrow (((j * q) \div p) * p) < (j * q)$

THEOREM: leq-quot-wins1

$(\text{prime}(p)$   
 $\wedge (\neg \text{divides}(p, q))$   
 $\wedge (j \leq (p \div 2))$   
 $\wedge (j \neq 0)$   
 $\wedge (q \neq 0))$   
 $\rightarrow (((j * q) \div p) \leq \text{wins1}(j * q, \text{mults}(q \div 2, p)))$

DEFINITION:

wins2( $a, n, p$ )

= **if**  $n \simeq 0$  **then** 0  
     **elseif**  $(n * p) < a$  **then**  $n$   
     **else** wins2( $a, n - 1, p$ ) **endif**

THEOREM: leq-wins2

$$(\text{wins2}(a, n, p) * p) \leq a$$

THEOREM: leq-wins1-n

$$\text{wins1}(a, \text{mults}(n, p)) \leq n$$

THEOREM: leq-wins1-wins2

$$\text{wins1}(a, \text{mults}(n, p)) \leq \text{wins2}(a, n, p)$$

THEOREM: leq-wins1

$$(\text{wins1}(a, \text{mults}(n, p)) * p) \leq a$$

THEOREM: leq-wins1-quot

$$(p \neq 0) \rightarrow (\text{wins1}(a, \text{mults}(n, p)) \leq (a \div p))$$

THEOREM: equal-quot-wins1

$$\begin{aligned} & (\text{prime}(p) \\ & \wedge (\neg \text{divides}(p, q)) \\ & \wedge (j \leq (p \div 2)) \\ & \wedge (j \neq 0) \\ & \wedge (q \neq 0)) \\ & \rightarrow (\text{wins1}(j * q, \text{mults}(q \div 2, p)) = ((j * q) \div p)) \end{aligned}$$

THEOREM: equal-wins-plus-quot-list

$$\begin{aligned} & (\text{prime}(p) \\ & \wedge (\neg \text{divides}(p, q)) \\ & \wedge (q \neq 0) \\ & \wedge (j \neq 0) \\ & \wedge (j \leq (p \div 2))) \\ & \rightarrow (\text{wins}(\text{mults}(j, q), \text{mults}(q \div 2, p)) = \text{plus-list}(\text{quot-list}(j, q, p))) \end{aligned}$$

THEOREM: gauss-corollary

$$\begin{aligned} & (\text{prime}(p) \wedge \text{prime}(q) \wedge (p \neq 2) \wedge (q \neq 2) \wedge (p \neq q)) \\ & \rightarrow (\text{res1}(p \div 2, q, p) = \text{residue}(q, p)) \end{aligned}$$

THEOREM: residue-quot-list

$$\begin{aligned} & (\text{prime}(p) \wedge \text{prime}(q) \wedge (p \neq q) \wedge (p \neq 2) \wedge (q \neq 2)) \\ & \rightarrow ((\text{residue}(q, p) = \text{residue}(p, q)) \\ & \quad = \text{even3}(\text{plus-list}(\text{quot-list}(p \div 2, q, p)) \\ & \quad \quad + \text{plus-list}(\text{quot-list}(q \div 2, p, q)))) \end{aligned}$$

THEOREM: all-non-zerop-mults

$$(p \neq 0) \rightarrow \text{all-non-zerop}(\text{mults}(n, p))$$

THEOREM: empty-intersect-mults-1

$$\begin{aligned} & (\text{prime}(p) \wedge \text{prime}(q) \wedge (p \neq q) \wedge (i < q) \wedge (j < p)) \\ & \rightarrow ((i * p) \notin \text{mults}(j, q)) \end{aligned}$$

THEOREM: empty-intersect-mults

$$\begin{aligned} & (\text{prime}(p) \wedge \text{prime}(q) \wedge (p \neq q) \wedge (i < q)) \\ \rightarrow & (\neg \text{listp}(\text{intersect}(\text{mults}(i, p), \text{mults}(p \div 2, q)))) \end{aligned}$$

THEOREM: equal-plus-quot-list-wins

$$\begin{aligned} & (\text{prime}(p) \wedge \text{prime}(q) \wedge (p \neq q)) \\ \rightarrow & (\text{plus-list}(\text{quot-list}(p \div 2, q, p)) \\ & = \text{wins}(\text{mults}(p \div 2, q), \text{mults}(q \div 2, p))) \end{aligned}$$

THEOREM: law-of-quadratic-reciprocity

$$\begin{aligned} & (\text{prime}(p) \wedge \text{prime}(q) \wedge (p \neq q) \wedge (p \neq 2) \wedge (q \neq 2)) \\ \rightarrow & ((\text{residue}(q, p) = \text{residue}(p, q)) = \text{even}((p \div 2) * (q \div 2))) \end{aligned}$$

## Index

- a-winner-every-time, 12
- all-distinct, 4, 9
- all-distinct-comp-list, 4
- all-distinct-comp-list-1, 4
- all-distinct-reflect-list, 9
- all-distinct-reflect-list-1, 8
- all-distinct-reflect-list-10, 9
- all-distinct-reflect-list-2, 8
- all-distinct-reflect-list-3, 8
- all-distinct-reflect-list-4, 8
- all-distinct-reflect-list-5, 9
- all-distinct-reflect-list-6, 9
- all-distinct-reflect-list-7, 9
- all-distinct-reflect-list-8, 9
- all-distinct-reflect-list-9, 9
- all-lesseqp, 3, 8
- all-lesseqp-reflect-list, 8
- all-lesseqp-reflect-list-1, 8
- all-non-zerop, 3, 8, 12, 14
- all-non-zerop-comp-list, 3
- all-non-zerop-mults, 14
- all-non-zerop-reflect-list, 8
- all-squares, 2
- all-squares-1, 1
- all-squares-2, 2
  
- bounded-comp-list, 3
- bounded-complement, 3
  
- comp-list, 3–6
- comp-list-closed-1, 4
- comp-list-closed-2, 4
- comp-list-fact, 4
- complement, 2–4
- complement-is-unique, 3
- complement-of-complement, 3
- complement-off, 3
- complement-works, 3
  
- delete, 5, 11
- diff-mod-1, 6
  
- divides, 1–6, 8–11, 13, 14
- double-reflect, 7
  
- empty-intersect-mults, 15
- empty-intersect-mults-1, 14
- equal-length-perm, 5
- equal-losses-wins, 12
- equal-plus-quot-list-wins, 15
- equal-quot-wins1, 14
- equal-wins-plus-quot-list, 14
- equals-have-same-parity, 11
- euler-1, 2
- euler-1-1, 2
- euler-1-2, 2
- euler-1-3, 2
- euler-1-4, 2
- euler-1-5, 2
- euler-1-6, 2
- euler-1-7, 2
- euler-2, 6
- euler-2-1, 6
- euler-2-2, 6
- euler-2-3, 6
- euler-2-4, 6
- even, 15
- even3, 11, 14
- even3-diff, 11
- even3-even, 11
- even3-plus, 11
- even3-rem, 11
- even3-rem-reflect, 11
- even3-times, 11
- exp, 2, 5–8, 10
  
- fact, 4, 7–9
  
- g0219, 2
- gauss-corollary, 14
- gauss-lemma, 10
  
- intersect, 12, 15
- inverse, 2



law-of-quadratic-reciprocity, 15  
 length, 5, 6, 8, 12  
 length-mults, 12  
 length-positives, 5  
 length-reflect-list, 8  
 leq-n-wins1, 12  
 leq-quot-times, 13  
 leq-quot-times-2, 13  
 leq-quot-wins1, 13  
 leq-quot-wins1-1, 13  
 leq-quot-wins1-2, 13  
 leq-times-quot, 13  
 leq-wins1, 14  
 leq-wins1-n, 14  
 leq-wins1-quot, 14  
 leq-wins1-wins2, 14  
 leq-wins2, 14  
 losses, 12  
 losses1, 12  
  
 monotone-quot, 13  
 monotone-quot-induction, 13  
 monotone-wins1, 12  
 mults, 12–15  
  
 no-self-complement, 3  
 non-zero-complement, 3  
 numberp-remainder, 8  
  
 perm, 4, 5, 11  
 perm-plus-list, 11  
 perm-plus-list-1, 11  
 perm-positives-comp-list, 4  
 plus-list, 10, 11, 14, 15  
 plus-mod-1, 8  
 plus-mod-2, 8  
 plus-reflect-list, 11  
 plus-x-x-even, 9  
 positives, 3–5, 10, 11  
 prime, 2–6, 8–11, 13–15  
  
 quot-list, 10, 11, 14, 15  
 quot-quot-induction, 13  
 quotient-plus-1, 5  
  
 reflect, 6–9  
 reflect-commutes-with-times-1, 6  
 reflect-commutes-with-times-2, 7  
 reflect-list, 6–9, 11  
 rem-diff-times, 6  
 rem-list, 10, 11  
 rem-quot-list, 10  
 rem-reflect-list, 8  
 rem-reflect-list-1, 7  
 rem-reflect-list-2, 7  
 rem-reflect-list-3, 7  
 rem-reflect-list-4, 7  
 rem-reflect-list-base-case, 8  
 remainder-lessp, 10  
 res1, 6, 8, 10, 11, 14  
 res1-quot-list, 11  
 res1-rem-1, 10  
 res1-rem-1-1, 10  
 res1-rem-2, 10  
 residue, 1–4, 6, 10, 14, 15  
 residue-quot-list, 14  
  
 squares, 1, 2  
 squares-off, 3  
 sub1-length-delete, 5  
 subsetp, 3  
 subsetp-positives-comp-list, 3  
  
 times-comp-list, 5  
 times-comp-list-1, 4  
 times-comp-list-2, 5  
 times-comp-list-3, 5  
 times-comp-list-4, 5  
 times-comp-list-5, 5  
 times-exp-fact, 7  
 times-list, 4, 5, 7–9  
 times-mod-4, 4  
 times-reflect-list, 9  
 two-even, 10  
  
 win-some-lose-some-1, 12  
 win-some-lose-some-2, 12  
 wins, 12, 14, 15  
 wins1, 11–14  
 wins2, 13, 14