1 Problem statement

Implement the algorithm for solving systems of congruences.

2 Algorithms

2.1 Modular inverse

Brute force algorithm

Require: $gcd(a, n) = 1$
Ensure: $b \cdot a \equiv 1 \mod n$
procedure MOD_INVERSE(a,n) ▷ Complexity: $\theta(2^n)$
for i=1,n **do**
if $a \cdot i \equiv 1 \mod n$ **then**
mod_inverse ← i
end if
end for
end procedure

2.2 Euclid's algorithm

Ensure: $bc = gcd(a, b)$
procedure EUCLID(a,b) ▷ Complexity: $\theta(n)$
if $a == 0$ **then**
euclid ← b
end if
if $b == 0$ **then**
euclid ← a
end if
while $a > 0$ **do**
temp ← a
a ← $b \mod a$
b ← temp
end while
euclid ← a
end procedure

2.3 Solve system of congruences using Chinese remainder theorem

Ensure: $gcd(n_i, n_j) = 1 \forall i, j = 1, r$

procedure SOLVE(a,n) ▷ Complexity: $\theta(\log^2(N))$

N $\leftarrow n_1 n_2 ... n_r$

solve $\leftarrow \sum_{i=1}^{r} a_i \frac{N}{n_i} (\frac{N}{n_i}^{-1} \mod n_i) \mod N$

end procedure