Finding inverses:
The Extended Euclidean Algorithm

- Inverses exists if **e** and **m** do not have any common factor.
- To find $\mathbf{e^{-1}}$ (inverse of e) such that $\mathbf{ee^{-1} = 1 \bmod m}$ we can use the Extended Euclidean Algorithm.
  - Before doing so it is instructive to look at the Euclidean algorithm.

GCD's and the Euclidean Algorithm

- The *greatest common divisor* (GCD) of two integers n_1 and n_2, not both zero, is the largest integer that divides n_1 and n_2.
- It is denoted **gcd(n_1,n_2)**.
- **Example**: gcd(30, 15) = 15

  gcd(30, -12) = 6
- We can calculate the gcd using Euclidean algorithm.

Euclidean Algorithm

1) Divide the larger number by the smaller and retain the remainder.
2) Divide the smaller original number by the remainder, again retaining the remainder.
3) Continue dividing the prior remainder by the current remainder until the remainder is zero, at which point the last (non-zero) remainder is the greatest common divisor.

- **Example**: gcd(84,49).

  84/49 ➔ remainder 35.

  49/35 ➔ remainder 14.

  35/14 ➔ remainder 7.

  14/7 ➔ remainder 0.

Therefore gcd(84,49)=7.

Extended GCD for integers

The Extended GCD Theorem for Integers states:

Given integers n_1 and n_2, not both zero, there exist integers a and b such that

$$gcd(n_1,n_2)=a*n_1+b*n_2$$

- These integers are not necessarily unique though.
- **Example**:

  gcd(15,12) = 3 = (+1)*15+(-1)*12

  = (+1-12)*15+ (-1+15)*12

  = (-11)*15+(+14)*12

If gcd(n_1, n_2)=1 then it means that we can find the inverses **n_1 mod n_2** and **n_2 mod n_1**.

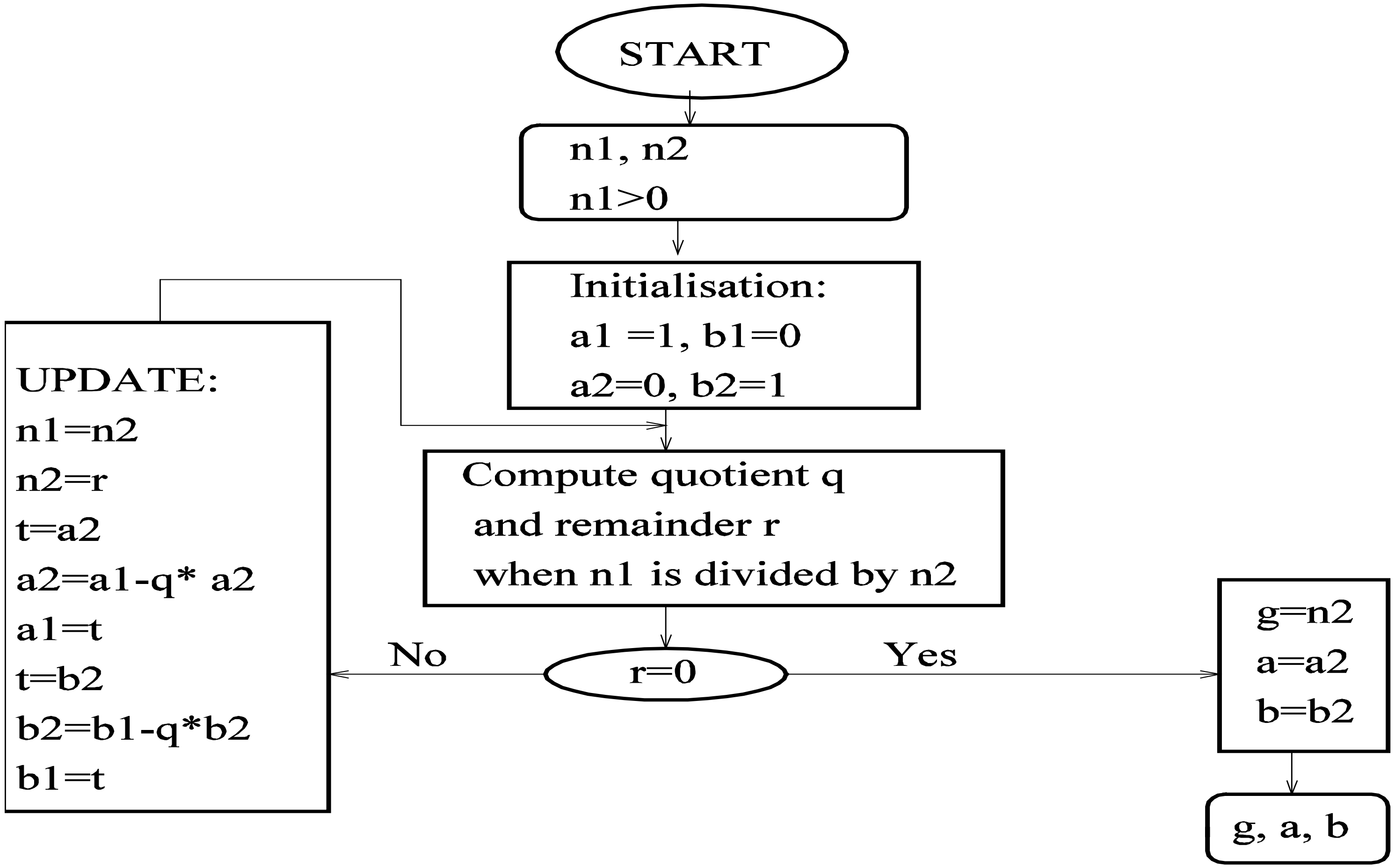
$$gcd(n_1,n_2)=a*n_1+b*n_2=1$$

- **Example**:

  gcd(65,14) = 1 = (-3)*65+(14)*14

  ➔ 14*14=1 (mod 65)

- The Extended Euclidean algorithm calculates a, b and $g=gcd(n_1,n_2)$ such that $g=a*n_1+b*n_2$.



Find gcd(39,11) and a,b, s.t 39a+11b=gcd(39,11)

	n_1	n_2	r	q	a_1	b_1	a_2	b_2
Initialise	39	11	6	3	1	0	0	1
	11	6	5	1	0	1	1	-3
	6	5	1	1	1	-3	-1	4
	5	**1**	0	5	-1	4	**2**	**-7**

gcd(39,11)=1

1=39*2+11*(-7)

a = 2, b = -7