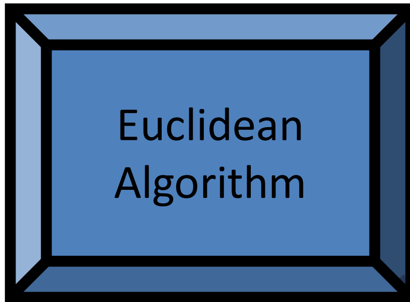
GCD and Optimization Problem

Euclidean Algorithm

m , n → 

→ gcd(m,n)

integer euclid(pos. … s. integer n)

$x = m, y = n$

while($y > 0$)

$r = x \bmod y$

$x = y$

$y = \mathrm{r}$

return x

Euclidean Algorithm. Example

gcd(33,77):

Step	$r = x \textbf{ mod } y$	x	y
0	–	33	77

Euclidean Algorithm. Example

gcd(33,77):

Step	$r = x \textbf{ mod } y$	x	y
0	–	33	77
1	33 **mod** 77 = 33	77	33

Euclidean Algorithm. Example

gcd(33,77):

Step	$r = x$ **mod** y	x	y
0	–	33	77
1	33 **mod** 77 = 33	77	33
2	77 **mod** 33 = 11	33	11

Euclidean Algorithm. Example

gcd(33,77):

Step	$r = x$ **mod** y	x	y
0	–	33	77
1	33 **mod** 77 = 33	77	33
2	77 **mod** 33 = 11	33	11
3	33 **mod** 11 = 0	11	0

Euclidean Algorithm. Example

gcd(244,117):

Step	$r = x$ **mod** y	x	y
0	–	244	117

Euclidean Algorithm. Example

gcd(244,117):

Step	$r = x$ **mod** y	x	y
0	–	244	117
1	244 **mod** 117 = 10	117	10

Euclidean Algorithm. Example

gcd(244,117):

Step	$r = x$ **mod** y	x	y
0	–	244	117
1	244 **mod** 117 = 10	117	10
2	117 **mod** 10 = 7	10	7

Euclidean Algorithm. Example

gcd(244,117):

Step	$r = x$ **mod** y	x	y
0	–	244	117
1	244 **mod** 117 = 10	117	10
2	117 **mod** 10 = 7	10	7
3	10 **mod** 7 = 3	7	3

Euclidean Algorithm. Example

gcd(244,117):

Step	$r = x$ **mod** y	x	y
0	–	244	117
1	244 **mod** 117 = 10	117	10
2	117 **mod** 10 = 7	10	7
3	10 **mod** 7 = 3	7	3
4	7 **mod** 3 = 1	3	1

Euclidean Algorithm. Example

gcd(244,117):

Step	$r = x$ **mod** y	x	y
0	–	244	117
1	244 **mod** 117 = 10	117	10
2	117 **mod** 10 = 7	10	7
3	10 **mod** 7 = 3	7	3
4	7 **mod** 3 = 1	3	1
5	3 **mod** 1=0	1	0

By definition → 244 and 117 are rel. prime.

Euclidean Algorithm Correctness

The reason that Euclidean algorithm works is gcd(x,y) is not changed from line to line. If x', y' denote the next values of x , y then:

$$\gcd(x',y') = \gcd(y, x \mathbf{\ mod\ } y)$$
$$= \gcd(y, x + qy) \quad \text{(the useful fact)}$$
$$= \gcd(y, x) \quad \text{(subtract } y \text{ -multiple)}$$
$$= \gcd(x,y)$$

Optimization Problem

- In mathematics and computer science, an **optimization problem** is the problem of finding the *best* solution from all feasible solutions. Optimization problems can be divided into two categories depending on whether the variables are continuous or discrete.
- An optimization problem with discrete variables is known as a **combinatorial optimization problem**. In a combinatorial optimization problem, we are looking for an object such as an integer, permutation or graph from a finite (or possibly countable infinite) set.

Thank You !!!