# Answers of exercises in the slides

# Section 1.1 – Introduction to design of algorithms

**Exercise 1**

**Find GCD(31415, 14142) by applying Euclid’s algorithm**

GCD(31415, 14142) = GCD(14142, 3131)

= GCD(3131, 1618)

= GCD(1618, 1513)

= GCD(1513, 105)

= GCD(105, 43)

= GCD(43, 19)

= GCD(19, 5)

= GCD(5, 4)

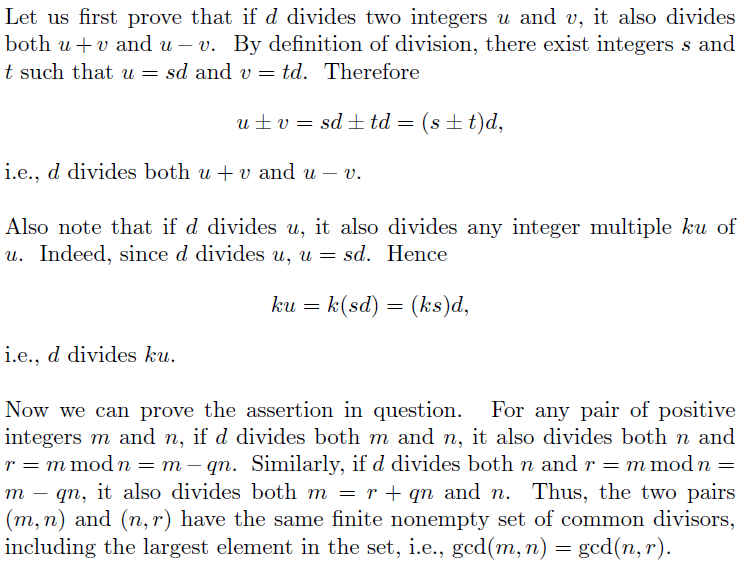
= GCD(4, 1)

= GCD(**1**, 0)

= 1

**Exercise 2**

**Prove the correctness of gcd(m, n) = gcd(n, m mod n) for the correctness of Euclid’s algorithm**



**Exercise 3**

**Design an algorithm to solve the quadratic equation**

**ALGORITHM** Quadratic(a, b, c)

**Input**: Three coefficients a, b, and c (real numbers, suppose that a ≠ 0)

**Output**: The root(s) of the equation E: ax2 + bx + c = 0 if they exist.

delta = b2 – 4ac

**if** delta > 0

print The roots of E are: *x*1, *x*2

**if** delta = 0

print The root of E is *x*1

**else**

print E has no root in R

**Exercise 4**

**Design an algorithm to check whether a list is a set.**

**ALGORITHM** isSet(A[0..n-1])

**Input**: An array A of n elements.

**Output**: True if A is a set, false otherwise. //True if A does not have duplicates

**for** i 🡨 0 to n -2 do

**for** j 🡨 i+1 to n -1 do

**if** A[i] =A[j] return false

**return** true

**Hints to Exercises 5 & 6**

**Exercise 5**: You can merge the two sets and then remove the duplicates.

**Exercise 6**: Create a new empty list C. Use a nested loop to search for any element A[i] from the 1st list A in the second list B. If A[i] is found in the second list, then add it to the new list C at a given index k.