The University Nottingham Ningbo China

Centre for English Language Education

Introduction to Algorithms(CELEN086)

Problem Sheet –3

Topics: Recursive algorithm; GCD; Prime number; Helper function

1. Write a recursive algorithm called **isEven()** that checks if a positive integer is even or not. It should return True if the number is even and False if the number is odd.

   Trace your algorithm for isEven(9).

2. Write a recursive algorithm called **sumDigits()** that takes a positive integer and returns the sum of its digits. For example,

   sumDigits(11)=2, sumDigits(1942)=16.

   Trace your algorithm with the given examples.

3. Write a recursive algorithm called **numDigits()** that takes a positive integer and counts the number of its digits. For example

   numDigits(11)=2, numDigits(1942)=4.

   Trace your algorithm with the given examples.

4. Find the GCD of following numbers

   (a) gcd(2022,12345)

   (b) gcd(924,198)

   (c) gcd(234,385)

   You should trace both algorithms as introduced in Lecture 3, namely gcd() and Euclid().

5. Write a recursive algorithm called **power(x, n)** that computes x^n. Here x is a non-zero real number and n is a non-negative integer.
   Trace your algorithm for power(5,3).

6. Write a recursive algorithm called **fakeLog(x, y)** that takes two positive integers x, y and returns the largest integer k such that $x^k \leq y$. For example,
   fakeLog(4,19)=2, fakeLog(3, 28)=3, fakeLog(5,4)=0.

   Test your algorithm with the given examples.

7. Two numbers that have 1 as their GCD are called *co-prime*.

   Which pair of numbers in Q4 is co-prime?

8. Two prime numbers p and q are called *twin-prime* if $|p - q| = 2$. For example, (11,13), (17,19), (41,43) are all pairs of twin-prime numbers. Determine if the following pairs are twin-prime or not.

   (a) (59,61)

   (b) (127,129)

9. Fibonacci numbers, commonly denoted by F_n, form a well-known sequence. The following is a Fibonacci sequence:

$$1, 1, 2, 3, 5, 8, 13 \ldots$$

   This sequence is defined recursively by the formula

$$F_n = F_{n-1} + F_{n-2} \quad (n \geq 3).$$

   Write a recursive algorithm called **fib()** that takes a positive integer n and returns the n-th Fibonacci number in this sequence. For example,

   $fib(1) = 1, fib(3) = 2, fib(7) = 13$.Trace your algorithm for fib(6).

10. Write a recursive algorithm called **fibSum()** that computes the sum of first n Fibonacci numbers. For example,

$$fibSum(1) = 1, fibSum(3) = 4, fibSum(7) = 33.$$

    You can call the algorithm/function fib() directly as written in Q9.

11. Write a recursive algorithm called **fakeSqrt()** that takes a positive integer n and returns the largest positive integer k such that $k \times k \leq n$.

    Trace your algorithm for fakeSqrt(10).

12. As mentioned in Lecture 3, euclid algorithm to find GCD(Greatest Common Divisor), use similar idea to find the **LCM (Least Common Multiple)**that is the smallest positive integer that is divisible by two numbers. For example, LCM(6,8)=24.

    Design an algorithm to compute the **LCM** of two positive integers.