1. Tribonacci numbers are a sequence of numbers similar to Fibonacci numbers, except that a number is computed by adding the three previous numbers. The beginning of the sequence is 1, 1, 2, 4, 7, 13, 24, 44, 81,…Assuming that the index starts with 0, write a recursive java method that computes the nth Tribonacci number. [5 Marks]
2. Let f(n) be the number of asterisks printed by the method below for a given value of n. Derive f(n) as a function of n. [5 Marks]

static void myPrint(int n)

{

If(n>=3)

{

myPrint(n-1);

System.out.println(“\*\*\*”);

myPrint(n-1);

System.out.println(“\*”);

}

}

1. Write a recursive java method for binary search. You may assume that the sequence of data has already been sorted in ascending order. The method returns the index of the target in the sequence or -1 if the target is not found. [8 Marks]

/\*data: the data sequence

low and high: the lower and higher end of the searching range,

which means searching within data[low] and data[high]

key: target to search

\*/

int binarySearch(int[] data, int low, int high, int key)

{

//check the parameters

//If not satisfied, print an error message and return -1

}

1. Given a recursive method below,

Int produce(num)

{

If(num>10) return 10;

Else return num+produce(num+1)/2;

}

1. What is the output by calling produce(7)? [4 Marks]
2. Rewrite the method using iteration instead. [8 Marks]

int produce(num)

{

}

1. The number of operations executed by algorithms A and B is 40n2 and 2n3, respectively. Prove briefly that A is better than B, deciding the constant ‘c’ and n0, as defined for Big-O notation. [5 Marks]
2. Given an n-element array X of integers, Algorithm C executes an O(n)-time computation for each even number in X, and an O(logn)-time computation for each odd number in X. What are the best-case and worst-case running times of Algorithm C? [5 Marks]
3. R-6.4 Implement a method transfer(S,T) that transfers all elements from stack S onto stack T, so that the element that starts at the top of S is the first to be inserted onto T, and the element at the bottom of S ends up at the top of T. You are expected to use common methods of stack. [5 Marks]

void transfer(Stack<E>S, Stack<E>T)

{

}